2001年5月1日発行（毎月1回1日発行）第20巻第5号 昭和57年12月6日第三種郵便物認可

マイコン
わかる！動かせる！
プログラムが組める雑誌

BASIC Magazine

MAY.2001
5
定価
550Yen



第2弾 携帯電話で動くゲームを作ろう!!
「iアプリ」制作・・「隕石避けゲーム」に挑戦

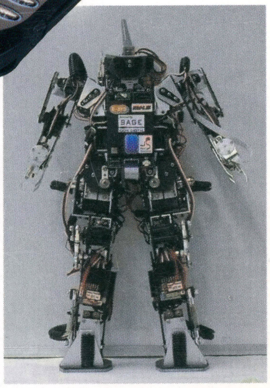


新連載
はじめようMicrosoft EXCELでのBASICプログラミング
PS2用BASIC「BASIC STUDIOパワフルゲーム工房」入門

ヒューマノイドロボット「Mk.5」の研究現場を見た！
青山学院大学理工学部情報テクノロジー学科 富山研

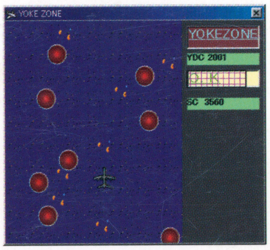


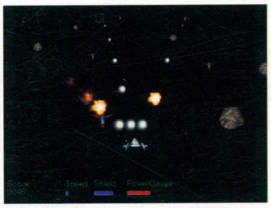


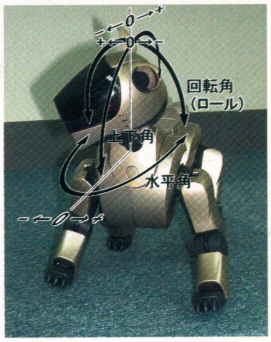


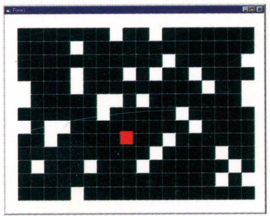
編集部の学校訪問記
千葉県立土気高校
長崎市立飽浦小学校










読者が作った作品をドーンと紹介
特選投稿プログラム

VisualBasic6.0	YOKE ZONE
VisualBasic6.0	VBもぐら
VisualBasic6.0	Gtool
VisualBasic6.0	BASIC Magazine投稿しようよ
VisualBasic5.0	森の守り番
VB6.0+DirectX	（超）レーザーシュートⅢ
VBScript+DirectX	グラデーション文字メーカー
HSP V2.5	@MAZE
Java（JDK1.1）	JavaSlot
PC-98	頭を使え！
FX-890P	いそげ宅急便
コンピュータ・ミュージック	
SC-88VL	寂寥の街Ⅱ
SC-8850	BEAT 4 US
SC-88	old-fashion



Visual Basic
Directx
ワンダーウィッチ

専用ホームページ http://www.basicmagazine.net/

ブロードバンド化が進むいま、データの維持管理はテープストレ

大容量、高速データ転送、しかも低価格で大切なデータを常に守るタンベルグデータ



Congraturation & Graduation
コングラジュエーションキャンペーン

ご卒業、ご入学、ご進学のお祝いに

ただいま、タンベルグデータでは生徒・学生・教育機関の方に向けたテープストレージ製品のコン・グラジュエーションキャンペーン実施中!!特別価格(アカデミックディスカウント)でご提供しています。
この期間中にお求めやすいタンベルグデータのテープストレージSLRシリーズ・NS20Proをぜひ勉強にお役立てください。

- 期間:2001年3月1日～4月30日
- 申し込み方法・お問い合わせは当社ホームページをご覧ください。

SLR40 容量40GB*
転送速度21.6GB*/時



Congraturation & Graduation 特別価格 ¥95,000※

- Windows NT/ 2000用に
ARCserve 2000トライアル版をセット。

注) SLR40/60はWindows95、98、Meには対応していません。
対応O/S、アプリケーションについてはホームページをご覧ください。

TANDBERG DATA

増大する
ージで!!

のテープストレージ。

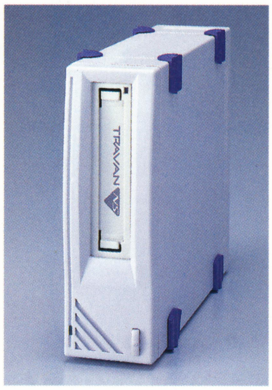
Tandberg
NS20Pro

TRAVAN Network Series NS™

Congratulation & Graduation 特別価格 ¥39,800※



最大容量20GB*、転送レート7.2GB*/時、高性能リニアテープレコーディング装置。SOHOやモバイルPCの大切なデータを守る強力な味方です。ワード、エクセル、パワーポイントなどのプレゼン資料、デジタルカメラから取り込んだ画像データなど、日々膨大になるデータを、FD、MOやCDに数GBのデータを書き込むには、ディスクの枚数も多くなり、入れ替え作業も必要になります。NS20Proならメディアー本で収まります。OSの入れ替え時も事前にバックアップを取っておけば、不慮のデータ損失時にも容易にデータリカバリが可能。セキュリティーツールとしても利用できます。プロ仕様をMO並の低価格でご提供。コンパクトで使いやすく、縦置きも可能なため場所もとりません。内蔵装置も取り揃えてあります。大手システムメーカ採用のテクノロジをあなたのPCライフに。



*ハードウェアによる圧縮方式を使用し、圧縮率を2：1と想定したときの値
●会社名、商品名は各社の商標または登録商標です。
※表記されている価格は内蔵タイプのみ(本体+データカートリッジ1巻)のキャンペーン価格です。

SLR60 容量60GB*
転送速度28.8GB*/時



Congratulation & Graduation 特別価格 ¥116,000※

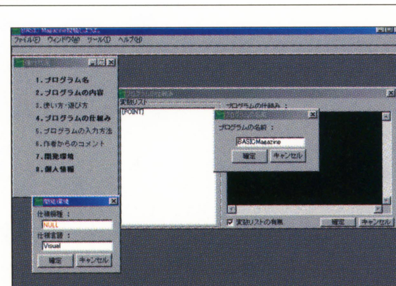
●Windows NT/ 2000用に
ARCserve 2000トライアル版をセット。

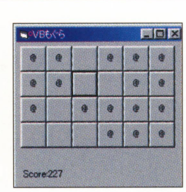
TapeStorage
信頼のストレージ、タンベルグデータ
タンベルグデータ

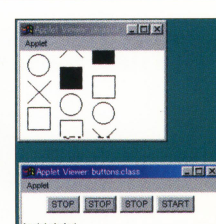
タンベルグデータ株式会社
〒104-0033 東京都中央区新川1-22-4 新川ニッテイアネックス7F
電話(03)5566-2871 FAX(03)5566-2875
http://www.tandberg.co.jp

マイコン BASIC Magazine CONTENTS

5
MAY 2001







専用ホームページ http://www.basicmagazine.net/

広告さくいん

第20巻 第5号 2001年4月25日印刷
2001年5月1日発行
定価550円（本体524円）
電波新聞社 2001 Printed in Japan
編集発行人：平山哲雄
東京都品川区東五反田1-11-15
郵便番号141-8715
電話 東京(03)3445-6111（大代表）
振替 （東京）00150-3-51961
印刷：奥村印刷(株) 製本：(株)堅省堂


maxell
PRO-X
Professional Excellent
COMPACT
disc
Recordable
MADE IN JAPAN

maxell

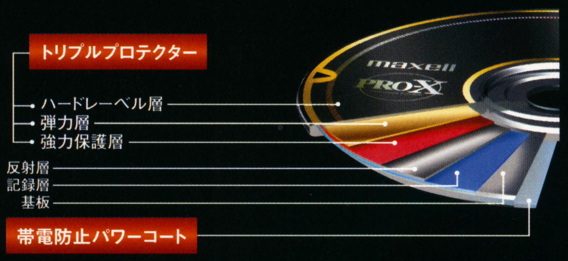
進化は次のステージへ。
プロフェッショナル仕様CD-R、
PRO-X誕生。

CD-R初*。帯電防止パワーコート&トリプルプロテクター採用、
大切なデータをディスクの記録面とレーベル面の両サイドから守ります。

PRO−X専用SSP(Super Smooth and Precise)スタンパを新開発、
MQシリーズを超える低エラーレートでさらなる信頼性を実現。

ディスクを外部衝撃から守るPRO−X専用SP (Shock Proof) スタウトケース。

*2000年9月現在。





SPスタウトケース



CDR80PRO.1TP

New CD-R
Professional Excellent
PRO-X

ARTDINK

PlayStation®2

"プレイステーション 2"で

キミの作ったゲームが"PS2"で動き出す。

『BASIC STUDIO パワフルゲーム工房』は、キミだけのオリジナルゲームが作れる「なんでもゲーム作成ツールズ」。
登場キャラクター、背景、サウンドを作成、プログラミングするだけで、キミの作品が"プレイステーション 2"で動き出すぞ。
しかも操作はとっても簡単。さあ、世界にひとつしかない、自分だけのオリジナルゲームを作って遊ぼう!

BASIC STUDIO・・・構成内容

3D MODELING

POLY-CRAFT(ポリクラフト)
ゲームに登場するキャラクターを
ポリクラフトで作成



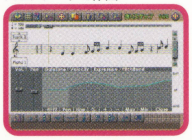
2D PAINTING

IMAGE-SKETCH(イメージスケッチ)
ゲーム内の背景を
イメージスケッチで描く



SOUND & MUSIC

SCORE-NOTE(スコアノート)
お気に入りの曲を
スコアノートで作曲



EZ-BASIC
(イージーベーシック)

BASIC

PROGRAMMING

最後にゲームを
プログラミングしよう!

こんなゲームも作れる!

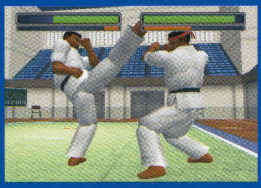


完成!!
Let's Play!!

作成したゲーム
は「B-SHELL」
を使えば簡単に
起動できるぞ!!

GAME

遊べるゲームも収録されて買ったその日から楽しめる!

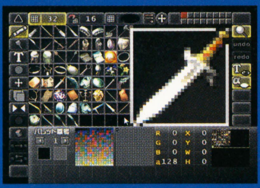


すぐに遊べるゲームが付いてくるから、作らなくても大丈夫。更に、改造も思いのままです。

収録タイトル
カルネージ・ヒート(シューティング)
徒手空拳(格闘)
フォール・アイランド(パズル)

SAMPLE

プロが作ったデータを、オリジナルゲームに使える!



実際に製品で使われたデータを、サンプル素材として収録。幅広いジャンルのCG・音楽データを組み合わせるだけで、すぐにゲーム作りが始められます。

わかりやすい入門マンガを同梱!



ベーシック言語に関しては、わかりやすい入門マンガで解説! キーボードの接続の仕方からはじまり、ベーシック言語での簡単なゲームの作り方まで、親切丁寧に解説。プログラムは初めてという方でも、肩の力を抜いて楽しむことができます。



ソニー製"プレイステーション 2"対応
プリンター『popegg』に対応。
popeggを使えば自分の作った
ゲームの名場面が簡単に印刷できます。

より快適に!!
USB接続のキーボード、マウス(共に別売り)の使用を推奨します。

BASIC STUDIO
・パワフルゲーム工房・

なんでもゲーム作成ツールズ
『BASIC STUDIO パワフルゲーム工房』
4月発売予定 標準価格**9,800円**(税抜)

 ※画面写真は開発中のものです。



『BASIC STUDIO パワフルゲーム工房』の詳しい情報は、http://www.artdink.co.jp/japanese/title/bs/index.html/まで

〒261-8701 千葉市美浜区中瀬2-6 WBG マリブウエスト22階 TEL.043-299-7890 株式会社アートディンク
URL http://www.artdink.co.jp/

中古通販

PC NET
ピーシーネット

在庫保有数日本一
(常時10,000台)

TEL.03-3733-1441

通販営業時間／月〜金 9:30〜19:00 土・日・祝日 11:00〜19:30

www.prins.co.jp/

楽天市場店 http://www.rakuten.co.jp/pcnet/ ネットプライスモール http://www.netpricemall.com/
ジャストネット http://emall.justnet.ne.jp/shop/pcnet/ ビッグローブショッピングワールド http://shopping.biglobe.ne.jp/

インターネットでは毎回お得な企画がいっぱい!! 他にもまだまだあります! 要チェックです!!

3日間保証セール だから安い!! 期間限定タイムセール 急げ!!

最新情報は電話かネットでチェック!!

思いがけない掘り出し物にめぐり逢えるかも!!

★★★ PC NET幕張店 OPEN記念 ★★★
特別プレゼント!
当選者発表!!

PC-VP16 CSAA1	FMV1NU 2WB0.5.6	FMV-NU 131.132.133共通
高橋 靖様 (栃木県小山市)	川那辺義弘様 (京都府京田辺市)	野崎達哉様 (福島県石川郡)

おめでとうございます
今後も当社のご利用を宜しくお願いいたします。

[安心の全品保証付] 1万円以上…1ヵ月保証 1万円以下…1週間保証

※本商品は添付品(OSを含む)は付属しておりません。
※クリーニングご希望の方、別途¥1,000が必要です。
※中古パソコン販売時の在庫定義が変わりました。
今後在庫とは、従来品に加え、動作上支障のないキズ・天板やけ・カバー等の欠品がある物や、キーボード等が補充(他メーカー含)されている商品も、「在庫」としてお取り扱いさせていただきます。

DOS/V

注文No.	商品	価格
U8444	MicroBook 200(液晶セット)(Akia)(Pen(200)/32MB/2GHD/CD/12.1TFT/SVGA/外装ヤケ/省スペース型/PCI×1/共有×1)	¥46,800
U1776	MicroBook ML5600TX/MP233(Akia)(MP(233)/64MB/3.2GHD/CD16/省スペース型/PCI×1/共有×1/USB×2)	¥54,800
U1507	Celebris GL6200(DEC)(PPro(200)/32MB/2GHD/CD8)…………	¥16,000
U8333	Celebris FX5200(DEC)(Pen(200)/32MB/1.2GHD/CD8/外装ヤケ・キズ/PCI×1/ISA×1/共有×1)	¥19,000
U2862	VENTURIS FX5166S(DEC)(Pen(166)/16MB/2.5GHD/CD6)…………	¥14,300
U1946	PC350(6587-JU4)(IBM)(Pen(166)/48MB/1.6GHD/CD6/ISA×2/共有×3/USB×2)	¥17,000
U2600	PC300 PL(6862-B2J/5J/6J)(IBM)(Celeron(333)/64MB/6.4GHD/CD32/LAN/PCI×2/ISA×1/共有×1/USB×2)	¥32,800
U1815	PC300 PL(6862-5JJ/KJ/LJ)(IBM)(Celeron(433)/64MB/6.4GHD/CD40/LAN/PCI×2/ISA×1/共有×1/USB×2)	¥36,800
U8453	PC300 GL(6288-39J/40J)(IBM)(Celeron(466)/96MB/10GHD/CD40/LAN/PCI×4/USB×2)	¥43,800
U1420	DESKPRO 2000/5166(COMPAQ)(Pen(166)/98MB/2.5GHD/CD8/外装塗装剥げ/PCI×2/ISA×2/共有×1)	¥18,800
U1932	DESKPRO 2000/5233X/3200CDS(COMPAQ)(MP(233)/32MB/3.2GHD/CD16/ミドルタワー型/PCI×2/ISA×1/共有×1/USB×2)	¥25,800
SH486	DESKPRO 4000S/5233X(COMPAQ)(MP(233)/32MB/2.1GHD/CD20)……	¥27,800
U2116	DESKPRO EP6300A(COMPAQ)(Celeron(300)/64MB/6.4GHD/CD32/タワー型/ディスクトップ型共用/PCI×3/ISA×2/USB×2/AGP×1)	¥28,800
U2266	OPTIPLEX GXMT5166(DELL)(Pen(166)/32MB/1.6GHD/CD6)…………	¥14,300
U8334	OPTIPLEX GXMT 5200(DELL)(Pen(200)/64MB/2.1GHD/CD4/タワー型/ISA×3/共有×2)	¥19,000
U1457	GATEWAY 2000(G6-200)(GATEWAY)(PPro(200)/32MB/2.5GHD/CD12/ミドルタワー型(44cm)/PCI×3/ISA×3/共有×1)	¥16,900
U8469	GATEWAY 2000(GP6-300)(GATEWAY)(PenII(300)/128MB/3.2GHD/CD24/タワー型(47cm)/PCI×3/ISA×1/共有×1/USB×2/AGP×1)	¥31,800
U1816	Vectra VL5/166-5(HP)(Pen(166)/16MB/1.6GHD/CD24/外装ヤケ)…………	¥14,300
U1245	Vectra VL6/333-7(HP)(PenII(333)/80MB/4.3GHD/CD24/PCI×1/ISA×1/共有×2/USB×2)	¥32,800
V0138	ZHP-D6588/802(HP)(PenII(400)/64MB/6.4GHD/CD32/100BT/付属品付◆Win98)	¥38,800
V0144	ZHP-D6582N/801(HP)(PenII(400)/128MB/6.4GHD/CD32/100BT/付属品付◆Win95)	¥49,800
U8395	VT516SR(T16SRX)(エプソン)(Pen(166)/64MB/1.2GHD/CD4/48MB増設/外装ヤケ/ミニタワー型/PCI×2/ISA×4)	¥14,800
U1265	FMV-516D5P(富士通)(Pen(166)/32MB/1.2GHD/CD4)…………	¥15,300
U8438	FMV-5166T5(FMV-516T5C1,5)(富士通)(Pen(166)/96MB/2.5GHD/CD8/LAN/64MB増設/外装ヤケ/タワー型/PCI×2/ISA×3/共有×1/USB×2)	¥19,800
U1846	FMV-520T4C1.5(FMV-5200T4)(富士通)(Pen(200)/32MB/2.5GHD/CD6/LAN/外装ヤケ/ミニタワー型/PCI×2/ISA×3/共有×1)	¥23,800
U2594	FMV-6200D7 FMV620D7C5,C7,D5,D7(富士通)(Ppro(200)/48MB/2.5GHD/CD8/LAN/外装擦り傷/PCI×1/ISA×2/USB×2)	¥14,000
U1725	FMV-521D9C5(富士通)(MP(200)/32MB/3.2GHD/CD16/外装キズ)…………	¥22,300
U8430	APRICOT MS660-F3C(三菱)(PII(300)/32MB/6.5GHD/CD24/DVD2/モデム/スピーカー・マイク・リモコン無し/ミニタワー型/PCI×2/ISA×3/共有×1/USB×2)	¥36,800
U1961	Millennia MME200(AN430TX-200M-MT)(MICRON)(MP(200)/64MB/3GHD/CD/PCI×3/ISA×2/共有×1/USB×2)	¥23,800

DOS/Vノート

注文No.	商品	価格
U6296	HiNote VP LTS5120(DEC)(Pen(120)/8MB/1.08GHD/SVGA/11.3TFT)……	¥22,800
U6355	ThinkPad 345C(2610-FJG)(IBM)(486DX4(75)/12MB/540HD/VGA/10.4TFT/8MB増設)	¥7,000
U3896	ThinkPad 365X(2625-3J9)(IBM)(Pen(120)/8MB/810HD/SVGA/10.4TFT)……	¥23,800
V0154	ThinkPad 600E(2645-4LJ)(IBM)(PII(366)/64MB/6.4GHD/CD24/56K/XGA/13.3TFT/付属品付◆Win98)	¥89,800
U3604	LATITUDE XPIP133ST(DELL)(Pen(133)/8MB/1.44GHD/VGA/PCカバー無し/11.3TFT)	¥29,800
U3939	FMV-50NLWS1,2,3共通(富士通)(Pen(100)/16MB/810HD/10.4TFT)…………	¥17,000
U3328	FMV-52NU2W(富士通)(Pen(120)/16MB/1GHD/SVGA/外装擦り傷/11.3TFT)	¥18,000
U3110	FMV-53NAWK(富士通)(Pen(133)/16MB/1GHD/SVGA/外装キズ/キーボード文字剥げ/PCカバー欠品/11.3TFT)	¥28,800
U3772	FMV-53NAWK(富士通)(Pen(133)/32MB/1GHD/SVGA/11.3TFT)…………	¥31,800
U6333	FMV-55NAXQ1・5(FMV-5150NA/X)(富士通)(Pen(150)/32MB/1.3GHD/CD6/XGA/16MB増設/12.1TFT/CD脱着式)	¥40,000
U6464	DynaBook PA1230C9(東芝)(Pen(120)/48MB/1.35GHD/SVGA/11.3TFT/32MB増設)	¥19,800
U3844	MEBIUS MN-7150(シャープ)(Pen(150)/24MB/1.35GHD/CD10/SVGA/外装塗装剥げ/11.3TFT)	¥43,800

バッテリーパックは消耗品につき保証対象外とさせていただきます。

98デスクトップ

注文No.	商品	価格
U2618	PC-9821V20(Pen(200)/32MB/2GHD/CD8/33.6kモデム)…………	¥27,800
U2630	PC-9821V200/M7C2(MP(200)/31.6MB/3GB/CD20/33.6K)…………	¥34,800
U2634	PC-9821V233/M3(MP(233)/32MB/3.2GB/CD20/Fax36.6K/ミニタワー型)…	¥38,800
U2764	PC-9821XA16/W30(Pen(166)/32MB/3GHD/PCM/CD8)…………	¥23,800
U1439	PC-9821XA20/W30(Pen(200)/48MB/3GHD/CD8/PCM/付属品付)…………	¥30,300
U1440	PC-9821XA20/W30(Pen(200)/64MB/3GHD/CD8/PCM/32MB増設/付属品付/◆Win95付き)	¥34,800
U2569	PC-9821XA20/W30R(Pen(200)/32MB/3GHD/CD8/PCM)…………	¥31,000
U1518	PC-9821XC13/S5A2(S5B2)(Pen(133)/32MB/1.6GHD/CD8/USB×2)……	¥18,800
U1636	PC-9821XC16/S5A3,S5B3(Pen(166)/32MB/1.6GHD/CD8/外装キズ)……	¥18,800
U1998	PC-9821XC200/S7A3,B3(MP(200)/64MB/2GHD/CD8/LAN/外装キズ・外装ヤケ/32MB増設済み/PCI×1/USB×2)	¥39,800
U1924	PC-9821XV20/W30(Pen(200)/64MB/3GHD/CD8/タワー型/付属品付◆Win95付き)	¥44,000
U1849	PC-9821RA20/N30(PPro(200)/96MB/4GHD/CD8/64MB増設)…………	¥40,800

98ノート

注文No.	商品	価格
U6324	PC-9821ND2/3(486DX2(50)/40MB/350HD/VGA/10.4DSTN)…………	¥12,300
U3293	PC-9821NE3/5(486DX2(50)/7.6MB/540HD/10.1TFT)…………	¥15,800
U3334	PC-9821NF/810(Pen(75)/7.6MB/810HD/PCM/9.5TFT)…………	¥19,800
U3084	PC-9821NS/340P(486DX2(50)/5.6MB/340HD/ピートスキー仕様/9.5TFT)…	¥11,800
U3205	PC-9821NX/3(486DX4(100)/7.6MB/350HD/10.1TFT)…………	¥18,800
U3135	PC-9821LT2/7A(486DX2(50)/8MB/720HD/サブノート/9.5TFT)…………	¥21,800
U3055	PC-9821LS13/D10.D10C2.D2(Pen(133)/32MB/1GHD/CD11/VGA/12.1DSTN)	¥49,800
U3269	PC-9821NB7/5(Pen(75)/8MB/540HD/PCM/外装キズ・ヤケ/10.4TFT)……	¥14,800
U6374	PC-9821NB7/5(Pen(75)/40MB/540HD/VGA/PCM/外装キズ/10.4TFT)……	¥17,000
U3204	PC-9821NB7/D5(Pen(75)/8MB/540HD/PCM/10.4DSTN)…………	¥11,800
U3077	PC-9821NB10/5(Pen(100)/8MB/540HD/PCM/外装キズ/10.4TFT)…………	¥17,800
U6397	PC-9821NB10/S8(Pen(100)/24MB/810HD/CD4/PCM/外装擦り傷/10.4TFT)	¥32,800
U6435	PC-9821NB10/S8(Pen(100)/80MB/810HD/CD4/VGA/PCM/64MB増設/10.4TFT)	¥37,800
U3695	PC-9821NB10/S10F(Pen(100)/16MB/1GHD/CD4/PCM/モデム欠品/10.4TFT)	¥30,800
U3323	PC-9821NA7/H7(Pen(75)/8MB/720HD/CD2/PCM 10.4TFT)…………	¥24,800
U6362	PC-9821NA12/S8(Pen(120)/24MB/810HD/CD4/VGA/12.1TFT)…………	¥32,000
U3386	PC-9821NA12/H10(Pen(120)/16MB/1GHD/CD4/PCM/SVGA/10.4TFT)…	¥32,000
U6419	PC-9821NA15/X14(Pen(150)/32MB/1.4GHD/CD8/XGA/バッテリーパック欠品/16MB増設/12.1TFT)	¥32,800
U0326	PC-9821NR13/D10B(Pen(133)/32MB/1GHD/CD8/PCM/外装塗装剥げ/FDケーブル欠品/12.1DSTN)	¥26,800
U0221	PC-9821NR13/D10A(Pen(133)/16MB/1GHD/CD8/PCM/12.1DSTN)……	¥34,000
U6471	PC-9821NR13/S1P(Pen(133)/48MB/1.4GHD/CD/SVGA/ピートスキー仕様/12.1TFT/CDオプション)	¥39,800
U6331	PC-9821NR15/S10(Pen(150)/32MB/1GHD/CD8/SVGA/キーボード文字剥げ/12.1TFT)	¥40,000
A0075	PC-9821NR15/S14F(Pen(150)/16MB/1.4GHD/CD10/36.6K/VGA/12.1TFT/付属品付)	¥49,800
U3629	PC-9821NR15/S14F(Pen(150)/16MB/1.4GHD/36.6K/SVGA/CD無し/12.1TFT)	¥33,800

バッテリーパックは消耗品につき保証対象外とさせていただきます。

MAC

注文No.	商品	価格
U0952	Performa 6310(PPC603e(100)/32MB/1.2G/CD)…………	¥8,800
U9202	PowerMac 6300/120(PPC603e(120)/16MB/1.2GHD/CD4)…………	¥9,800
U9276	PowerMac 7200/90(PPC601(90)/32MB/800HD/CD4)…………	¥9,800
U9281	PowerMac 7200/120(PPC601(120)/32MB/1.2GHD/CD4/外装ヤケキズ)	¥10,800
U9344	PowerMac 7300/180(PPC604e(180)/16MB/2GHD/CD12)…………	¥24,000
U9192	PowerMac 7500/100(PPC601(100)/49MB/1GHD/CD)…………	¥20,000
U9396	PowerMac 7600/120(PPC604(120)/49MB/1.2GHD/CD4/外装ヤケ)……	¥21,000
U9274	PowerMac 8100/100AV(PPC601(100)/32MB/1GHD/CD2/外装ヤケ/外装キズ)	¥12,800
U9175	PowerMac 8500/132(PPC604(132)/64MB/1.2GHD/CD)…………	¥28,800
SH483	PowerMac 8500/150(PPC604(150)/32MB/2GHD/CD4/付属品付)……	¥32,800
U9404	PowerMac 8500/180(PPC604e(180)/64MB/2GHD/CD8)…………	¥34,800
U9460	PowerMac 9500/120(PPC604(120)/48MB/1GHD/CD4)…………	¥34,800
U9301	PowerMac 9500/150(PPC604(150)/32MB/2GHD/CD4)…………	¥44,800
U9468	PowerMac 9500/200(PPC604e(200)/98MB/2GHD/CD8)…………	¥53,800
U9474	PowerMac G3/DT233(PPCG3(233)/64MB/4GHD/CD24/Zip付き[ZIPは動作保証無し])	¥48,000

ディスプレイ

注文No.	商品	価格
U4457	PC-DV17B1(NEC)(17マルチ:1280×1024max/3列15pin)…………	¥13,800
U4318	PC-KM173R2(NEC)(17マルチ:1280×1024max/0.28mm/外装ヤケ/3列15pin)	¥9,800
J0131	A6009(AXIL)(20マルチ:1280×1024/0.30mm/BNC欠/98・DOS/V)……	¥27,800
U4561	D1025TM(DELL)(17マルチ:1280×1024max/3列15pin)…………	¥11,800
U4446	D-1726T-HS(トリニトロン)(DELL)(17マルチ:1024×768max/外装ヤケ/3列15pin)	¥11,800
U4772	VRCX1-W3(digital)(21マルチ:1600×1200max/3列15pin)…………	¥35,800
U4132	7097-5B6(IBM)(17マルチ:1280×1024max/3列15pin)…………	¥9,800
U4353	CPD-17SF9(ソニー)(17マルチ:1280×1024max/0.25mm/外装ヤケ/3列15pin)	¥16,000
U4590	GDM-20SE1(ソニー)(20マルチ:1280×1024max/0.30mm/5BNC)……	¥36,000
U4493	FMV-DP978(17インチ)(富士通)(17マルチ:1024×768max/3列15pin)……	¥14,800
U4639	RD-17MX(三菱)(17マルチ:1280×1024max/3列15pin)…………	¥13,800
U4616	RD-17GXII(ダイヤモンド)(三菱)(17マルチ:1280×1024max/0.25mm/外装ヤケ/3列15pin)	¥15,000
U4469	RD-17V(ダイヤモンド)(三菱)(17マルチ:1280×1024max/0.25mm/3列15pin/直付)	¥14,800
U4369	MT-8617ES(飯山電機)(17マルチ:1600×1200max/0.25mm/3列15pin/BNC)	¥15,800
U4551	PC-DC3576(日立)(17マルチ:1280×1024max/3列15pin)…………	¥9,800
SH650	TX-D7F36-J(パナソニック)(17マルチ:1024×768max/3列15pin)……	¥11,000
U4460	CPD-GF200(GATEWAY)(17マルチ:1024×768max/3列15pin)……	¥13,000
U4522	VIVITRON 1100(GATEWAY)(20マルチ:1280×1024max/3列15pin)……	¥29,800
U0414	VX 1100(GATEWAY)(21マルチ:1600×1200max/3列15pin)…………	¥32,800

プリンタ

注文No.	商品	価格
SH543	PR-L2000X2(NEC)(A4/1500DPI:マルチライタープリンター/付属品付)……	¥28,800
SH530	LP-9200S(エプソン)(レーザー:A3対応/付属品付)…………	¥34,800

トナー等は消耗品につき標準添付されておりません。別途、ご確認下さい。

**商品の状態により価格の変動がある事がございます。
詳しくはTELかネット上でご確認ください。**

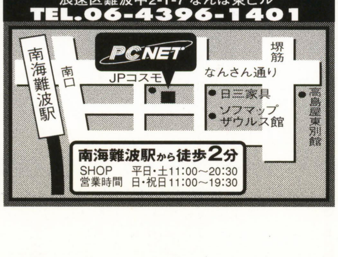
万一、品切れの際はご容赦ください。(3月15日現在)

こちらの在庫リストは通販専用となります。店舗在庫は各店舗へご来店の上ご確認ください。

秋葉原店
千代田区外神田6-5-12 借楽ビル新末広2F(1Fジョナサン)
TEL.03-5817-1441



大阪日本橋店
浪速区難波中2-1-7 なんば東ビル
TEL.06-4396-1401



幕張店
千葉県千葉市美浜区ひび野1-9



パシフィック ネット
株式会社パシフィックネット
本社 〒144-0051 東京都大田区西蒲田7-10-1
相互西蒲田ビル TEL.03-3733-1441 FAX.03-3733-1440
通販営業時間／月〜金 9:30〜19:00 土・日・祝日 11:00〜19:30

計測機器を各ショップのみで販売中。
詳しくは各ショップまでお問い合わせください。

iモードでショップ最新情報検索
http://www.prins.co.jp/i/

編集部の学校訪問記



▲元気のいい英語科2年生

国際理解教育とインターネットで，伝統づくりに教師と生徒が一丸に！

千葉市東端に位置する土気町，昭和の森公園をはじめ緑豊かな地区です。今月紹介するのは，昭和の森公園に隣接する土気高校（土屋博紀校長）。千葉市内の22番目の公立高校として昭和58年に創立された学校です。普通科のほか英語科があり，平成3，4年には「国際理解教育推進指定校」となるなど，国際理解教育に力を入れており，アメリカとオーストラリアの高校との姉妹校交流も盛んで，その成果を着実にあげています。

コンピュータへの取り組みも比較的早く，現在ではホームページの作成のほか，英検やワープロ検定取得のための学習や各授業などに幅広く利用されています。

千葉県立土気（とけ）高等学校

http://academic1.plala.or.jp/toke-hs/

「英語一般」の授業でホームページ作成

取材時は，英語科2年生の「英語一般」の授業を拝見しました。「FrontPage Express」を使い，ホームページの作成をするというものです。テーマは「英語で都市を紹介しよう」。実際訪れてみて思い出に残った町や憧れの町など，生徒それぞれの感性でページを作成していきます。それは，外国の町だったり，日本の観光地だったり，さまざまです。

事前に紙上でデザインを考えてきている生徒が多く，着々と進みます。写真や旅行ガイドブック，パンフレットなどを持ち込んでは，スキャナで取り込んだり，事前に用意されている観光名所の素材を使うなど，手際よくできていきます。

英語科ですので，英語の表現は得意分野。「FrontPageExpress」の使いかたや，写真の加工などで，あちこちから「先生～！」という声がかかります。英語科は女生徒の数が大半を締めており，とてもにぎやかです。

締め切りもしっかり決められており，3時間で（3回の授業の中で）完成させることになっています。出来の良い作品は，学校の公式ホームページでも紹介していくのだそうです。

ちなみに，「英語一般」の授業は，
1学期…Windowsの基礎・タイピング練習・MSWordの基本的操作
2学期…MSWordの応用・E-mailの使いかた（Hotmail）・HTMLタグを使ったホームページの作成練習
3学期…FrontPage Expressを使ってのホームページの作成（日本語・英語）
というカリキュラムになっています。



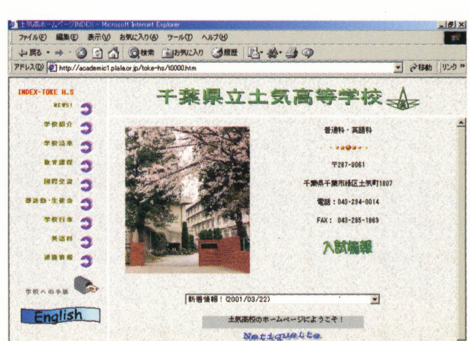
ホームページコンテストで最優秀賞に輝く

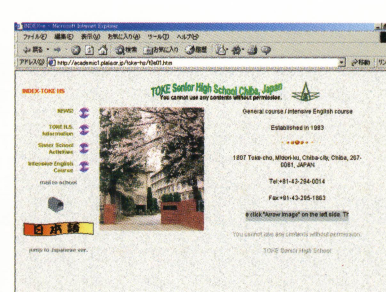
土気高校のコンピュータ導入は比較的早く，平成元年に25台設置と同時にコン





▲情報教育委員の松田理先生（左）と吉永和彦先生（右）。同校のコンピュータ教育にはかかせない存在





◀▲日本語と英語版が用意されているホームページ。ほとんど松田先生と吉永先生で更新しているという

ピュータ室（情報処理室）を設けました。平成10年の3月にはWindows95マシンに更新，富士通の「FMV5233T7M」40台を導入しました。現在はWindows98の「FMV6366DX2C」2台を加え42台を生徒用として活用しています。

平成11年に「ぷらら」が提供する学校向けサービス「えでゅけっと」に加入し，インターネットにも接続可能となり，同時にホームページも立ち上げ情報発信しています。千葉日報社が主催する「学校ホームページ・コンテスト」では，昨年（第1回）の最優秀賞に続き，今年（第2回）も優秀賞を受賞しています。

姉妹校交流が学習意欲を盛り上げる！

「特色ある学校づくり」の一環で，英語教育と国際理解教育を高めている同校ですが，アメリカのニコレ高校とオーストラリアのバリナ高校と姉妹校提携を結んでいます。毎年短期留学生の交換（受入れ・派遣）をメインにした国際交流活動を実践しています。

留学生は約3週間，姉妹校生徒宅にホームステイするというもので，そのようすは，インターネットを使い写真付きで学校に報告されます。

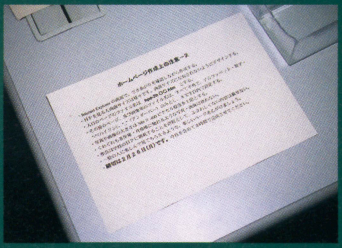


▲英語科の鳥江政弘先生









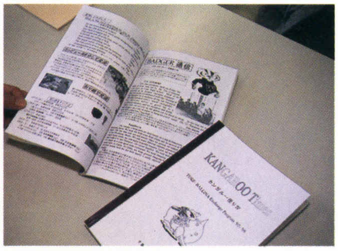




▶英語科2年生の「一般英語」では「英語で都市を紹介しよう」というテーマでホームページを作成していた。事前準備も万端で，スキャナも難なく扱い，さくさく進む



◀教室の壁に貼りだされているワープロ検定問題例と受験申し込み受付



◀アメリカとオーストラリアの高校との交換留学時に，メールで報告された内容は，冊子にまとめられている

「ぷらら」の学校向けサービス「えでゅけっと」

「えでゅけっと」とは，NTT東日本のグループ企業・ぷららネットワークスが運営するプロバイダ「ぷらら」が，1998年4月から提供している学校向けインターネット接続サービス。学校で手軽に利用してもらえるようにと，低めの価格設定や，教育上不適切と思われるサイトの閲覧を防止できるフィルタリング機能が好評で，2001年3月現在，全国約2,700校で利用されている。

「フレッツ・ISDN」，「フレッツ・ADSL」に対応した新メニュー「アカデミックライセンス・スーパー」もサービス開始された。

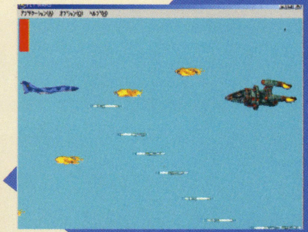
問合わせ先

＜ぷらら法人会員サービス担当＞
TEL：03-5954-7257
E-mail：school@plala.co.jp
「ぷらら」学校向けサービスホームページ
【ぷらら学園都市】
http://biz.plala.or.jp/school/

情報処理部

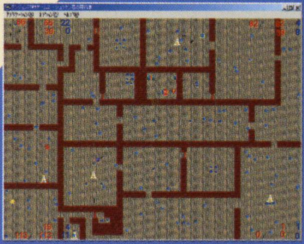


▶顧問の森藤茂男先生（左）と情報処理部員。まん中が部長の管くん





◀自作のDOS/V機。まだ完成前？



作品集（ゲームは主に「click & create」で作成されたもの）

パソコンでゲームを作ったり，MIDIを使って電子音楽を作ったり，プログラミングを中心に活動しています。また，生徒会予算や部活動振興費用を駆使し，DOS/V機の自作にも挑戦しています。主な活動発表の場は文化祭で，自作ゲームなどを公開しています。活動場所は物理教室。

千葉県立土気高等学校 〒267-0061 千葉県千葉市緑区土気町1807

※この記事は2月の取材によって構成されています。



▲6年生のみんな。パノラマ写真でどうぞ

編集部の学校訪問記

長崎市のコンピュータ教育研究モデル校として実績を築き上げる

長崎市立飽浦（あくのうら）小学校

長崎市立飽浦小学校（稲富義人校長）は，富国強兵期の明治８年に誕生した伝統ある学校です。現在，同校は長崎市教育委員会からコンピュータの研究校に指定され，パソコンを授業に活用してきました。今年中には待望のインターネット接続もはじまる予定です。

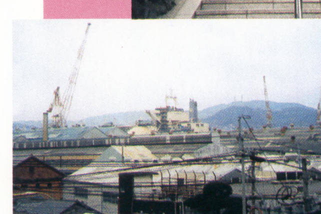
長崎は歴史，平和などいろんな側面を持った都市ですが，造船も有名です。飽浦小学校は，当時，造船所で働く人たちの子女のためにつくられた学校でした。飽の浦町には三菱重工の長崎造船所があり，小学校の校庭から見ることができます。「規律を守り，いささかも労を惜しまず，超重機の重きに耐うる心もて，学びの道にいそしまん」という校歌の歌詞が創立当時の気質を伝えています。



◀町並みから見た飽浦小学校校舎。



◀校舎正面。長崎といえば坂ですね



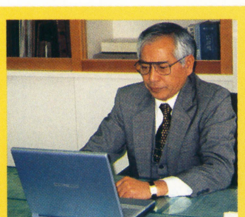
◀校舎の玄関から見える三菱の造船所。造船は長崎を支えてきた

「生きる力を育む情報教育の実践」を主題とした取り組み

飽浦小学校は，創立120周年を迎えた平成7年に新校舎が新築されました。そのとき，NECのPC9821CX2が20台，PC982xa10が2台導入され，それ以来，長崎市内のコンピュータ教育研究のモデル校として，グループ学習などでパソコンを活用してきました。

平成11年と12年は「生きる力を育む情報教育の実践」を主題とし，副主題に「学習の道具としてのコンピュータ活用を中心に」を設定して，コンピュータ教育を行ってきました。コンピュータの研究主任は4年生担当の山佐和久先生です。

「コンピュータを使いはじめて7年，1年生から6年生全員が使っています。普段あまり作文が好きではない生徒でも，パソコンは修正が自由ですから，パソコンだと安心して使っているようです。音楽も生徒それぞれの力にあわせたリズム学習ができます」と山佐先生はコンピュータ学習の効果を強調しています。



◀稲富校長先生。取材時は卒業式の祝辞を考えておられた



▲下小牧教頭先生。学校の実務を取り仕切っています



◀コンピュータ研究主任の山佐先生

「卒業を前にして」をテーマに音声付き画像作成に取り組む

編集部が訪問したクラスは卒業を間近に控えた6年生で，取材当日は黒岩敏博先生が国語を教えていました。教室はパソコンのあるコンピュータ室です。「卒業を前にして」というテーマでNECの「スーパーYUKI」というソフトの作図機能，録音機能などを使って音声つき画像を作るのが目的です。

6年生のみんなはグループごとに学校で思い出に残ったことを四つの場面に分け，イラストとセリフにして紙に書き出します。スーパーYUKIを使い，コンピュータでイラストをもとに作図，その図

にあわせてセリフや効果音を録音します。そうするとイラストにあわせて音声が流れるようになります。

授業は1時間弱ですが，黒岩先生はテキパキと授業をすすめていました。授業の最初で「4時間で全部完成しなければいけません。絵にかけられる時間は上限3時間まで。2時間で絵を描き，終わったグループは音声を入れてください」と時間を守ることの大切さを強調しています。そうなんです，大人になると時間制限と戦わなければいけないのです（そしてどれだけ多くの人が時間制限のため泣いているか……）！

黒岩先生は「コンピュータで絵を描くことは中学を卒業すればなかなか時間もなく，できないことです。ここで大切なのは時間が限られていますから，どこに力を入れてどこで手を抜くかを学ぶことです。この6年生たちもこれからホームページをインターネットで探す機会もあるでしょう。ただこだわりだけで作業をすると，ものすごい時間がかかります。だから，この子たちが，この授業で本当に伝えたいところに力を入れることを学んでほしいのです」と話していました。うーん，さすが黒岩先生。これは大人の用語で「効率」というのです（かぎカッコをつけた理由は大人になればわかります）。

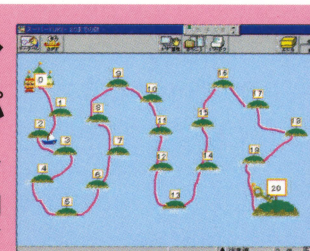
黒岩先生はまた「調べものをするにしても，なんでもかんでもインターネットを使うのではなく，新聞や本・雑誌も利用して調べてほしいですね」とこれから中学で学ぶみなさんにアドバイスしてくれました。

CATVインターネット導入も間近！今後にますます期待

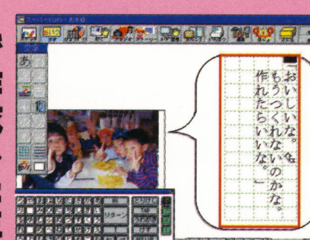
飽浦小学校にうれしいニュースが入っています。それは今年中にCATVのインターネットができるようになるということです。長崎市は市立全学校を対象に長崎ケーブルメディア（本社＝長崎市）の回線を使って「あじさいネット」と呼ばれるネットワークをつくることを決定しています。飽浦小学校でも現在職員室のパソコン1台がインターネットに接続していますが，これからは生徒も使えるようになるそうです。下小牧真一教頭先生は「先生全員がメールアドレスを持ち，学校からの通知はEメールでするようになるかもしれません」と話しています。

稲富義人校長先生は「今コンピュータを使っている総合的な学習という授業は，最終的には年105時間までに増える予定です。インターネットは情報収集，双方向通信で使われていくと思います」と予想します。ケーブルテレビのインターネットは電話回線に比べ通信速度が速く，動画や音声も送信しやすいのです。飽浦小学校の授業もこれから大きく変わるのではないでしょうか。

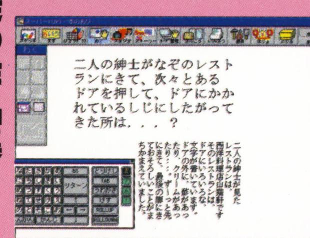
スーパーYUKIで作成した生徒の作品集



1年生 テーマ＝20までの数



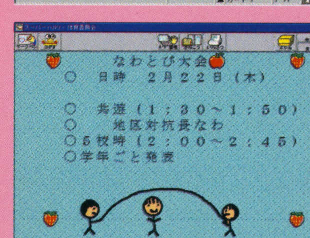
2年生 テーマ＝お手紙



3年生 テーマ＝本のおび



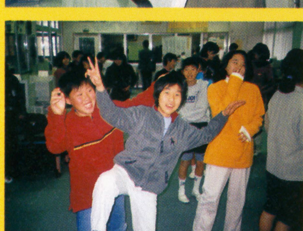
4年生 テーマ＝一つの花



体育委員会の告知



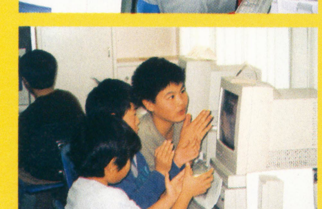
◀緩急自在の授業テクニックを見せる黒岩先生



◀ちょっとパチリ！彼らは将来、吉本興業志望とのこと



◀コンピュータルームは広く、グループ学習を意識した配置



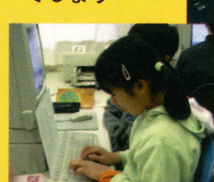
◀作品発表で一番にぎやかだったグループです



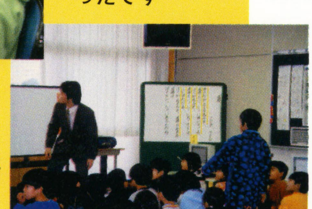
▲2月当時は6年生だったみんな。現在は新中学生として張り切っていることでしょう



▲彼女たちはキャンプファイヤの絵を描いていました。人物配置がうまかったです

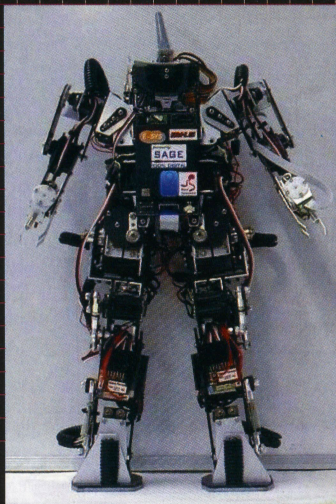


▲3年生の国語の授業風景



▶5年生の国語の授業風景

飽浦小学校　〒850-0063　長崎県長崎市飽の浦町17-1　メールアドレス　e30@nagasaki-city.ed.jp



▲「Mk.5」／1999年11月完成／全長356mm／重量1900g／制作費：70万円／開発期間20日／設計開発メンバー４名

編集部のエレクトロニック研究所訪問記

ヒューマノイド「Mk.5」（マーク・ファイブ）



青山学院大学理工学部情報テクノロジー学科
富山研究室（エンジニアリングシステム研究室「E-SYS」）
http://www-esys.me.aoyama.ac.jp/

同研究室のWEBサイトでヒューマノイド「Mk.5」を初めて見たとき，まずはそのカッコ良さに感激しました。それから動歩行をするという事実に驚き，これはぜひ自分の目で見て確かめたいと，青山学院大学世田谷キャンパス（東京都世田谷区）にある富山研究室……別名「エンジニアリングシステム研究室(E-SYS)」にお邪魔してきました。

富山研究室について

富山研究室の富山教授にお話をうかがいました。同大学でのロボットへの取り組みは，1988年に富山教授が同大学の理工学部機械工学科メカトロニクス研究室に着任されたのが始まりだそうです。

「ロボットは総合芸術」との言葉どおり，研究では多くの分野の技術を総合的に学べる環境作りを目指しているそうです。富山研究室は「情報テクノロジー学科」に所属していますが，「情報テクノロジー学科」はソフトウエアとハードウエアの融合を目的として，2000年4月に誕生した新しい学科です。

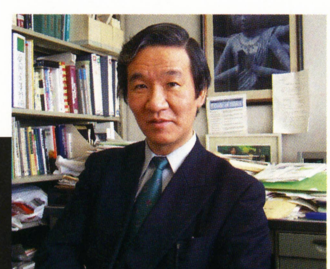
さらに富山教授は，プレゼンテーションの大切さも強調されていました。研究だけではなく，その研究内容を伝える能力の育成にも力を注がれているようです。

「Mk.5」誕生まで

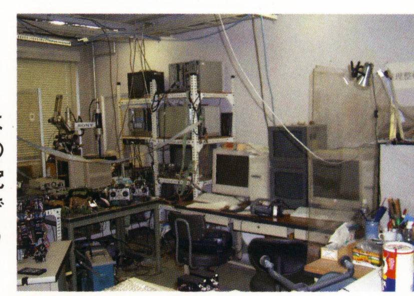
同研究室で行われている「パーソナルヒューマノイドプロジェクト」は古田さんと４～５名の研究メンバーによって1996年４月（ホンダのP2が発表される以前）からスタートしました。このプロジェクトでは，技術的な目標と完成時期が厳密に管理されました。さらに開発能率を上げるために，班単位で並行して研究する方法が取られました。

班は要素技術別に，動歩行・全身運動の制御を研究する「二足歩行制御班」，行動生成・知能を研究する「AI班」，そしてロボットの視覚を研究する「ビジョン班」の3班にわけられました。こうした個々の技術を蓄積する段階を，同プロジェクトでは「第1フェーズ」と呼んでいました。

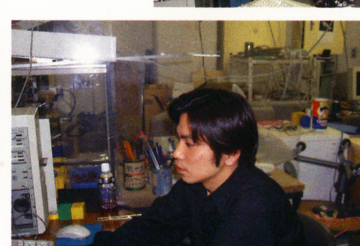
夏休みになると，研究は朝10:00から翌日の朝7:00まで（！）にもおよび，それが約2か月間続くようなハードなもの



青山学院大学
国際交流センター所長
理工学部情報テクノロジー学科
富山　健　教授

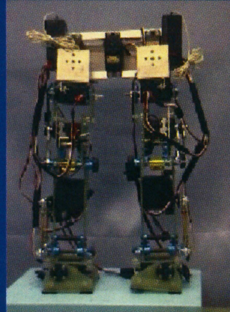


▶研究室のようす。大抵の加工なら研究室内の設備でできてしまうらしい

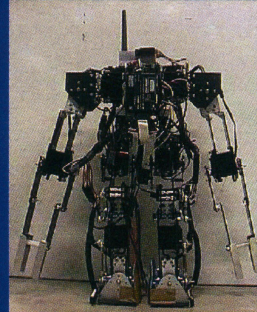


◀「ヒューマノイドプロジェクト」は古田さんと17～18名の学生とで研究が行われている（写真は博士前期過程1年の板橋さん。担当はAI班）

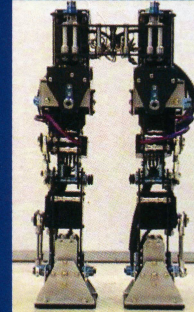
「E-SYS ヒューマノイドプロジェクト」の歴史



◀「Mk・1」／1997年11月

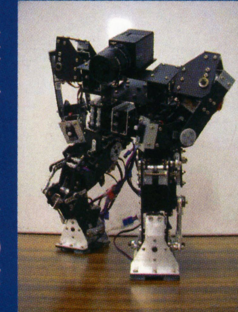


◀「Mk・2」／1998年3月



◀「Mk.3」／1998年6月完成／全長303mm／重量1400g／制作費45万円／開発期間５日／設計開発メンバー２名

▶「Mk.4」／1998年11月完成／全長363mm／重量2000g／制作費50万円／開発期間30日／設計開発メンバー４名

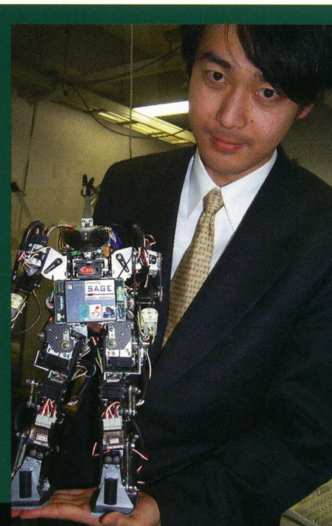


＊開発日数は20時間以上＝1日で計算

になったそうです（現在も泊まりこみで研究されていました）。その苦労が実り，2000年5月にはヒューマノイド「Mk.5」が完成しました。

「Mk.5」は全長35cmと非常に小型ですが，小型化することで，コストを抑えて効率よく研究ができるそうです（制作費は70万円）。「Mk.5」は2000年に行われたRoboCupメルボルン大会で，衝撃的なデモンストレーションを果たし，その名は世界にも知られるようになりました。

同プロジェクトは2000年度から，各要素を統合・検証する「第2フェーズ」という段階に移行しています。



青山学院大学理工学部
情報テクノロジー学科 助手
工学博士　古田貴之さん

富山教授をして「新しいタイプの研究者」と言わしめるのが，「ヒューマノイドプロジェクト」の立案者でプロジェクトリーダでもある助手の古田さんです。

小学5年生の頃に「お茶運びロボット」を自作していたほどのロボット好きで，中学生の頃，足に大きな怪我をしたときに，古田さんは「足イス」（「車イス」でなくて）という装置を考案していたとのことです。

「昔，ベーマガ読んでましたよ」との言葉どおり，かなりのパソコン・マニアらしく，MZ-2000に始まり，PC-1251やX68000でアセンブラを覚えたそうです（ロボットのアニメにもくわしいことは秘密らしい）。

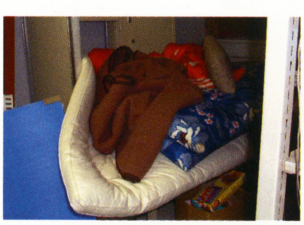
1996年4月に同大学の情報テクノロジー学科の助手に就任し，念願のテーマであった「ヒューマノイドプロジェクト」を立ち上げたそうです。

2001年4月以後，古田さんは「科学技術振興事業団 ERATO 北野共生システムプロジェクト（文部科学技術省の研究機関）」に在籍されていますが，今も富山研究室では客員研究員として指導にあたられています。

つぎなるフェーズに

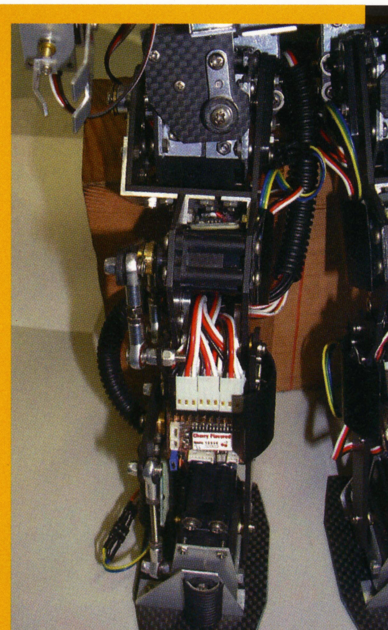
この研究室では厳密なクオリティコントロール，企業からの協力，PR活動などが活発に行われ，新しい研究スタイルが実践されているようです。キャンパス移転も近く予定され，「大学が変わる」という印象を強く受けました。

現在，17～18名の研究メンバーを抱えるほどに成長した「E-SYS　ヒューマノイドプロジェクト」は「第3フェーズ」に突入しようとしています。そこでは，人をサポートしたり，安全性を考えたヒューマノイドを研究されるようです。最終的なゴールは，古田さんのお話の中にあった「人を幸せにするロボット」かと思われますが，とりあえずは「Mk.6」が楽しみです。



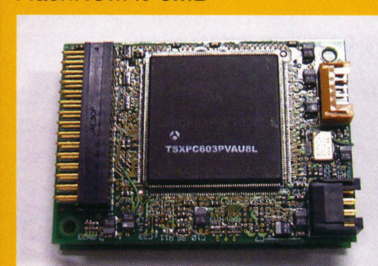
◀泊まりこみの研究が多いため布団が積まれている。これが「不夜城」と言われるゆえんか

「Mk・5」のココが凄い！

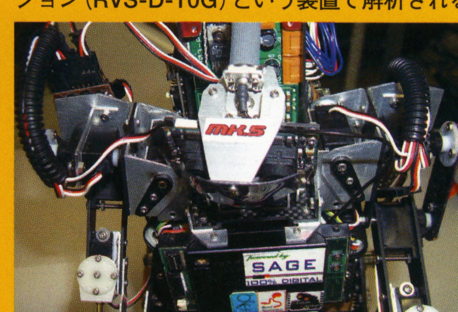


▲フレームは軽くて上部な「カーボン-FRP」（炭素繊維を樹脂で固めた素材）を使用。各関節のモータ制御やセンサ情報処理をつかさどるPICモジュールが全身4か所に配置され，体内LANシステムによってSAGE-CPUモジュールとPICモジュール間とのデータ通信を行っている

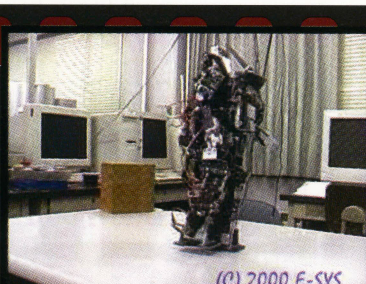
▼「Mk.5」から初めて搭載されたオリジナルのCPUボード「SAGE-CPUモジュール」。プロセッサはPowerPC 603e（200MHz），RAMは64MB，FlashROMは8MB



▼頭部のCCDカメラ2個によって三次元の空間認識が可能。画像は無線LANを介してIMAPビジョン（RVS-D-10G）という装置で解析される



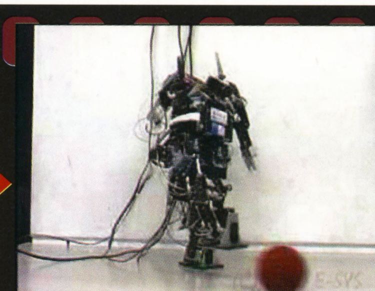
「Mk・5」ムービー（撮影・富山研究室）



▲歩行



▲その場で旋回



▲ボールを認識してキック

＊世田谷キャンパスは2003年4月開学の新キャンパス（神奈川県相模原市）に移転する予定です。

携帯電話で動くゲームを作ろう!!

君にもできる「iアプリ」制作

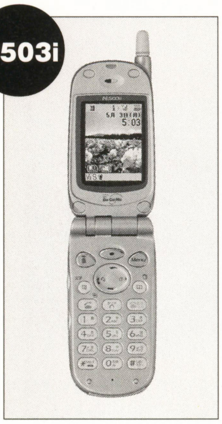


第2弾 「隕石避けゲーム」に挑戦！

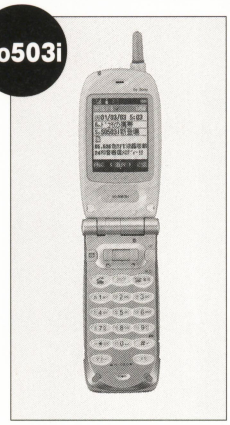
ついにすべての主要メーカーからｉアプリ対応携帯電話機がリリースされました。そして，今まで以上に活気づくｉアプリ市場ですが，とくに懐かしの名作ゲームの復活が目を引きます。オールドゲーム・ファンにとってはうれしいことではないでしょうか。　（松原拓也）

新機種情報!!

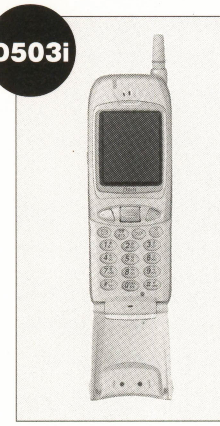
「F503i」「P503i」に続いてフリップタイプの新機種が続々と登場しました。　＊サイズは折りたたみ時のもの



◀「デジタル・ムーバN503i HYPER」（日本電気製）／オープン価格（実売32,800円）／高さ93mm ×幅 48mm ×厚さ 22mm／質量約98g／連続通話時間約135分



◀「DoCoMo by Sony SO503i」（ソニー製）／オープン価格（実売 34,800円）／高さ 98mm ×幅49mm ×厚さ 28mm／質量約 115g ／連続通話時間約140分／24和音着信音，カラーTFT液晶65,536色表示



◀「デジタル・ムーバD503i HYPER」（三菱電機製）／オープン価格（実売30,800円）／高さ129mm ×幅46mm ×厚さ 17mm／質量 約81g／連続通話時間 約130分／ATOK Pocket搭載

ｉアプリ新作ゲーム情報!!

ピコピコゲームファン必見のソフトが勢ぞろいです。

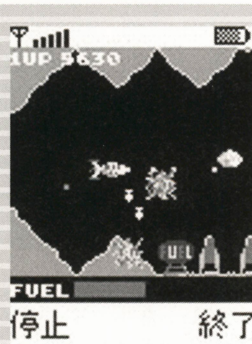


「キャラゲーランド機動戦士ガンダム」

（月額150円）バンダイネットワークス

モビルスーツによる対戦ゲームや，キャラクターを使ったタイピング・ゲーム等，数種類のゲームを用意。ラインナップは随時追加予定

Ⓒ創通エージェンシー・サンライズ

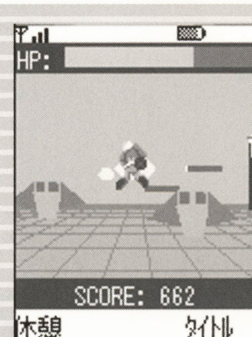


「e－ゲーセン」

（月額300円）コナミ

「ビシバシジャンプ」「ランキングパラダイス」「BemaniAudition」「スクランブル」（！）が遊べてしまう配信サービス。

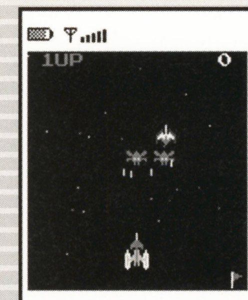
同じくコナミの「コナミデラックスパック」（月額300円）では「けっきょく南極ちょい冒険」「プチプーヤン」なども配信中



「ソニックカフェ」

（月額300円）セガ

名作「スペースハリアー」がラインナップに新登場。F503iでは動きが少々重たかったので，新機種での処理速度が気になるところ。音楽はパートが短いながらも見事に再現されている



「アプリキャロット」

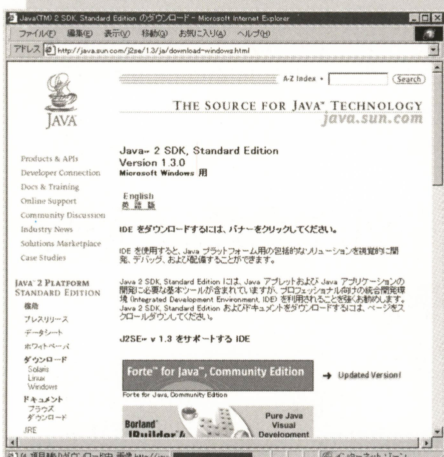
（月額300円）ナムコ

「パックマン」「ギャラクシアンmini」「ワニワニパニック」などゲーム・センターでおなじみのゲームが遊べる

ｉアプリの開発環境を作ろう

ｉアプリの開発には「Java SDK」「J2ME CLDC」『iモード拡張クラスライブラリ」「携帯電話エミュレータ」が必要です。では，ｉアプリの開発方法を簡単におさらいしてみましょう。

＊ さらにくわしい開発方法については2001年4月号の特集記事を参照してください。



①サンマイクロシステムズのWEBサイト（http://java.sun.com/）から「Java 2 SDK Standard Edition 1.3.0」をダウンロードします。インストールすると“C:¥jdk1.3”というディレクトリにJavaコンパイラが展開されます。

②さらに同サイトから「Java 2 Micro Edition, Connected Limited Device Configuration (J2ME CLDC) Version 1.0」をダウンロードします（ユーザー登録が必要です）。ダウンロードしたZIPファイルは解凍して，“c:¥j2me_cldc”ディレクトリに展開します。

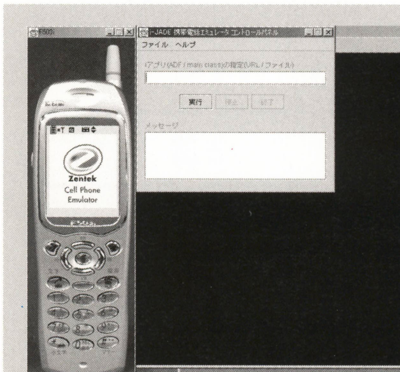
③NTTドコモのi-mode関連WEBサイト（http://www.ntt docomo.co.jp/i/）から，ｉモード拡張クラスライブラリのドキュメントをダウンロードしてます。

④ゼンテック・テクノロジー・ジャパンのWEBサイト（http://www.zentek.com/i-JADE/）から『i-JADE Lite』をダウンロードします（ユーザー登録が必要です）。インストールすると“c:¥i-jade”に携帯電話エミュレータとクラスライブラリが展開されます。

⑤“c:¥i-jade”フォルダの下に“classes”というフォルダを作成します。

⑥“c:¥i-jade”フォルダにある“i-jade-*.jar”（“*”部分は使用する機種）をZIP対応の解凍ユーティリティ（Explzhなど）を使って，“c:¥i-jade¥classes”に展開します。

⑦“c:¥j2me_cldc¥bin¥api¥classes”にある“java”フォルダを“c:¥i-jade¥classes”にコピーします。



⑧Javaのソースファイルは Java コンパイラ“javac.exe”でクラスファイルに変換します。それから，“preverify. exe”によって，クラスの検証を行います。検証済みのクラスは“jar. exe”で圧縮して最終的に一つのJARファイルになります。

compile.bat - メモ帳

```
echo off
path C:¥jdk1.3¥bin;C:¥j2me_cldc¥bin;
if EXIST class¥NUL goto compile
md class
:compile
javac -g:none -classpath c:¥i-jade¥classes; *.java -d class
if ERRORLEVEL 1 goto err
preverify -classpath c:¥i-jade¥classes; class
if ERRORLEVEL 1 goto err
jar cvfM test.jar -C output .
echo 正常終了しました
goto end
:err
echo エラーで終了しました
:end
```

⑨JARファイルは携帯電話エミュレータを使ってパソコン上で実行できます。JARファイルはJAMファイルというテキストファイルによって参照されます。

test.html - メモ帳

```
<HTML>
<HEAD>
<TITLE>test</TITLE>
</HEAD>
<BODY>

ｉアプリ<BR>

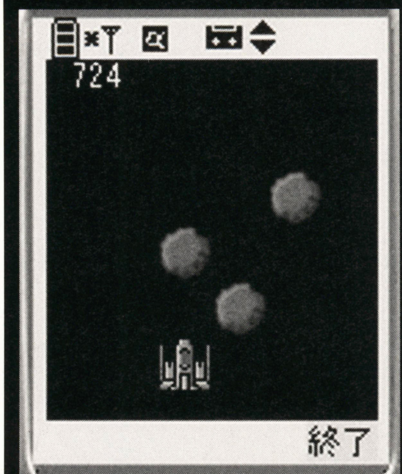
<OBJECT declare id="b" data="http://www.basicmagazine.net/i/test.jam"
type="application/x-jam">
test</OBJECT><BR>
<A ijam="#b" href="test.html">テストアプリ</A>

</BODY>
</HTML>
```

⑩実機でｉアプリを実行するには，JARファイルとJAMファイルとHTMLファイルをサーバにアップロードしておく必要があります。HTMLファイルの書式はｉアプリ専用で，JAMファイルのリンク先を記述しておきます。携帯電話を使ってHTMLファイルを開き，ｉアプリのリンク先をクリックすると，携帯電話内にｉアプリがダウンロードされます。一度ダウンロードしたｉアプリは，以後，回線の接続なしで実行することができます（スタンドアロン版の場合）。

つぎのページでは実際にプログラムを作ってみましょう。

サンプルプログラム「隕石避けゲーム」



▲宇宙船を操って隕石を避けるというシンプルながら熱いゲーム

この「隕石避けゲーム」は宇宙船を操って，画面上方から降りそそぐ隕石をかわすゲーム（別名「チープインパクト」）です。得点に応じてちゃんと難易度が上がるようになっています。

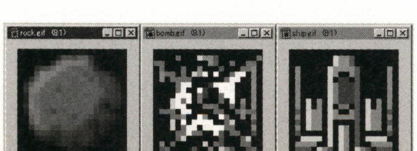
入力方法

まずメモ帳やエディタでリスト1～3のプログラム・リストを打ち込みます。それから，「隕石」「爆発」「宇宙船」をそれぞれ20×20ピクセルの大きさで描き，“rock.gif” “bomb.gif” “ship.gif” というGIF画像で “output” フォルダに保存します。あとは，“compile.bat(4月号参照)” などを使ってビルドしてください。

技術的な解説

（リスト中の矢印と連動しています）

① **“getImage”メソッドでGIFファイルを読み込みます。読み込まれた画像データは “Image” クラスに格納され，“drawImage”メソッドによって表示**



▲作成した3枚のGIF画像。大きさは20×20ピクセル

《リスト1》“GameApp.java”

```
// サンプルゲーム「チープインパクト」"GameApp.java"  by 松原拓也

import com.nttdocomo.util.*;  /*クラスのインポート*/
import com.nttdocomo.ui.*;

public class GameApp extends IApplication
{
    public void start()
    {
          GameScreen screen = new GameScreen(); /*キャンバスの作成*/
          Display.setCurrent(screen);  /*キャンバスのカレント表示*/
    }
}
```

《リスト2》“GameScreen.java”

```
/* 「チープインパクト」キャンバス "GameScreen.java" */

import com.nttdocomo.ui.*;
import com.nttdocomo.io.*;
import java.util.*;  // クラスをインポート

public class GameScreen extends Canvas implements Runnable {

    final int TIPW = 20; /* 常数:キャラの一辺のピクセル数 */
    final int MAPW = 6;  /* 常数:画面一辺のチップ数 */
    final int SEQSTART = 0;
    final int SEQDEAD = 1;
    final int SEQEND = 50;
    int iMx,iMy;        /* 主人公座標 */
    Thread myThread;  /* スレッド */
    Image imgMy;        /* イメージ:主人公 (宇宙船) */
    Image imgBomb;      /* イメージ:主人公 (爆破) */
    Image imgRock;      /* イメージ:隕石 */
    int iSeq;           /*シーケンス*/
    int iScore ,iNando;  /* スコア、難易度 */
    StringBuffer strMap = new StringBuffer();   /* マップデータ */
    Random rnd = new Random();   /* 新しい乱数列を作成 */

    public Image getImg(String sName) { /* イメージ読み込み関数 */
        MediaImage mi = MediaManager.getImage(sName);  ◀…… ❶
        try {
              mi.use(); /* イメージの読み込み */
        } catch (ConnectionException e) {
        }
        return(mi.getImage());
    }

    /*文字列位置関数*/
    public int fncPos(int iX ,int iY){
        return(iX + (iY * MAPW));
    }

    public void subInit(){    /*ゲームデータのクリア*/
        int iCnt;

        iMx = MAPW/2;  /* 主人公座標 */
        iMy = MAPW-1;
        iSeq=SEQSTART; /*シーケンス初期化*/
        iScore=0;       /*スコア0点*/
        iNando=0;       /*難易度0*/
        for(iCnt=0 ;iCnt<MAPW*MAPW ;iCnt++)
            strMap.append(' ');  /*マップデータのクリア*/
    }

    public GameScreen() {    /* コンストラクタ */
        setSoftLabel(Frame.SOFT_KEY_2, "終了");
        imgMy   = getImg("resource:///ship.gif"); /* GIF画像読み込み */
        imgBomb = getImg("resource:///bomb.gif");
        imgRock = getImg("resource:///rock.gif");

        myThread = new Thread(this); /* スレッド作成 */  ◀…… ❷
        myThread.start();  /* スレッド開始 */

        subInit();  /*ゲームデータのクリア*/
    }

    public void paint(Graphics g) {   /* ペイントメソッド */  ◀…… ❸
        int iGx,iGy,iPos;

        g.lock();  /* 画面のロック */
        g.setColor(g.getColorOfRGB(0, 0, 0));
        g.fillRect(0, 0, getWidth(), getHeight() ); /*宇宙*/

        /* マップの描画 */
        for(iGy=0 ;iGy<MAPW ;iGy++ ) {
             for(iGx=0 ;iGx<MAPW ;iGx++ ) {
                  if(strMap.charAt(fncPos(iGx ,iGy)) =='*') {
                        g.drawImage(imgRock ,iGx*TIPW ,iGy*TIPW);
                  }
             }
        }
```

《リスト3》“cinpact.jam”

```
AppName = チープインパクト
KvmVer = CLDC-1.0
AppClass = GameApp
AppVer = 1.0
PackageURL = cinpact.jar
AppSize = 3528
LastModified = Fri,16 Mar 2001 23:30:20
```

```
        g.setColor(g.getColorOfRGB(255, 255, 255));
        g.drawString(Integer.toString(iScore), 10, 10);  /*スコア表示*/

        if(iSeq == 0)
            g.drawImage(imgMy ,iMx*TIPW ,iMy*TIPW);   /* 主人公の描画 */
        else
            g.drawImage(imgBomb ,iMx*TIPW ,iMy*TIPW); /* 主人公爆発 */

        g.unlock(false);  /* ロック解除 */
    }

    public void run() {  /* スレッド実行 */
        int iKey;
        int iTx,iTy;
        char cTmp;

        for(;;) {  ◀…………❹

            try {
                Thread.sleep(50);
            } catch(InterruptedException ignore) { }

            iKey=getKeypadState();        /* 主人公の移動 */

            if(iSeq == SEQSTART) { /*シーケンスによる分岐*/

                iScore++;  /*スコアをプラス*/
                if((iScore % 100)==0)
                    iNando++;     /*徐々に難易度をプラス*/

                /*主人公の移動*/
                if((iKey==(1 << Display.KEY_LEFT))&&(iMx > 0)) {
                    iMx--;
                }
                if((iKey==(1 << Display.KEY_RIGHT))&&(iMx < (MAPW-1))) {
                    iMx++;
                }

                /*当たり判定*/
                if(strMap.charAt(fncPos(iMx,iMy)) == '*')
                    iSeq = SEQDEAD;  /*隕石に接触してゲームオーバー*/
            }else{
                iSeq++;  /*爆破中。一定時間見せたら最初に戻る*/
                if(iSeq > SEQEND)
                    subInit();  /*ゲームデータのクリア*/
            }

            /*画面スクロール*/  ◀…………❺
            for(iTy=(MAPW-2) ;iTy>=0 ;iTy-- ) {
                for(iTx=0 ;iTx<MAPW ;iTx++ ) {
                    cTmp = strMap.charAt(fncPos(iTx,iTy));
                    strMap.setCharAt(fncPos(iTx,iTy+1) ,cTmp);
                }
            }

            for(iTx=0 ;iTx<MAPW ;iTx++ ) {   /* マップ継ぎ足し */
                if( Math.abs(rnd.nextInt() % 100) > iNando)  ◀…………❻
                    cTmp=' ';
                else
                    cTmp='*';  /* 隕石出現!! */

                strMap.setCharAt(fncPos(iTx,0) ,cTmp);
            }

            if(iKey==(1 << Display.KEY_SOFT2)) {
                IApplication.getCurrentApp().terminate(); /*終了*/
            }
            repaint(); /*再描画*/  ◀…………❼
        }
    }
}
```

されます。

② アクション・ゲームはリアルタイムで独立した処理を行う必要がありますから，スレッドを作成します。

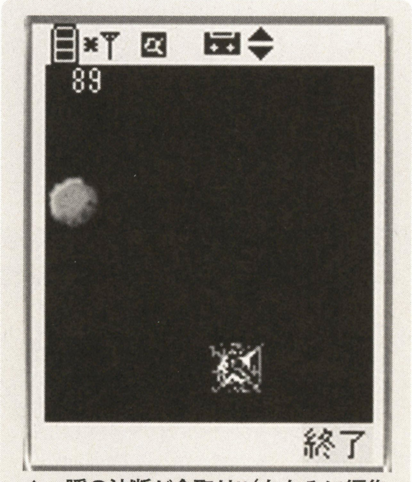
③ “paint” メソッドでは直接の画像更新にかかわる，すべての処理を行います。まず，背景を黒く塗りつぶし，それから仮想マップをもとに隕石を描き，それから座標に合わせて宇宙船（自機）を描きます。隕石の所在は，“StringBuffer” という文字列（クラス）で作った仮想マップに格納されます。文字列の特定の部分を抜き出すためには “charAt” を使います。

④ スレッド内の処理です。最初の “for(;;)”を見て，「無限ループはマズいのでは？」と思ってしまう人がいるかもしれませんが，独立したプロセス（処理）なので問題ありません。

⑤ 仮想マップを下方向に縦スクロールさせます。順序的に下の行から上に向かってコピーしますので，Y座標はマイナス値のステップでループさせています。文字列の部分変更は “setCharAt” を使います。

⑥ 乱数を発生させて隕石の出現処理を行います。乱数は“Random”クラスの “nextInt” メソッドで発生させます。“Math.abs(rnd.nextInt() % 100)” と記述することで，0～99までの乱数を発生させることができます。

⑦ 更新した仮想マップを画面に反映させるため，“paint”イベントを発生させます。



▲一瞬の油断が命取り!!（ちなみに編集部内でのハイスコアは883点）

今回掲載したサンプル・プログラムはマイコンBASIC MagazineのWEBサイトからダウンロードすることができます。ぜひ試してみてください。

http://www.basicmagazine.net/

パソコンレクチャー 2001

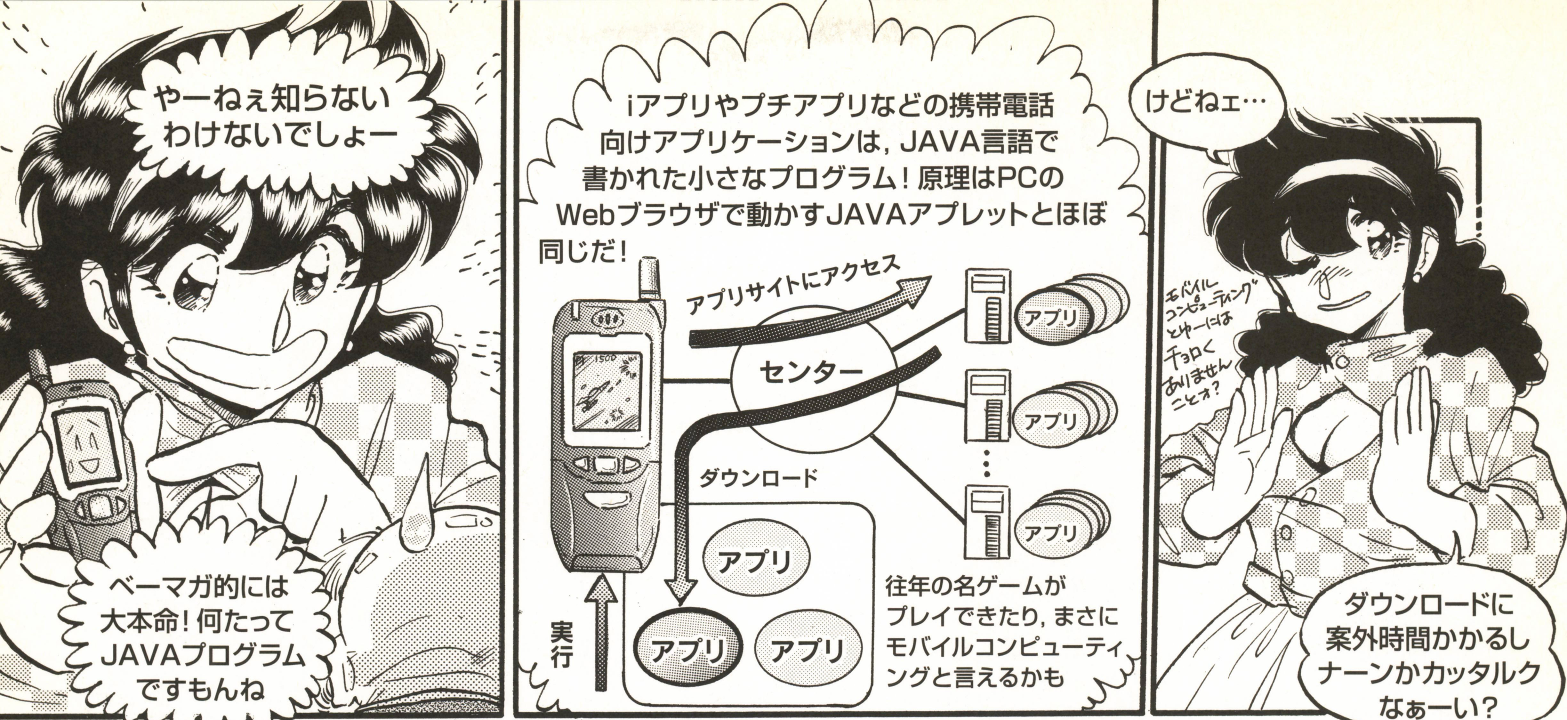
モバイル編①
「モバイル新世紀！ IMT-2000実現!!」

by
くりひろし





ほぇー
じ，時代は変わるもんですね〜〜〜
Dr.Dがケータイでメールねェ…
何じゃい今時年寄りだってケータイくらい当たり前で使うぞよ
ま，たしかにベーマガ創刊のころなんて携帯電話はなかったな
移動電話といえば自動車電話とかのことで…持ってる人は商売やってる人くらいかなぁ
ん〜なつかしの自動車電話…
ほんの20年もたたないうちに今や小学生でも5人に1人がケータイ持ちじゃ
ナマイキじゃよな
とーおもう
モバイルなんてのもそもそもコトバがなかったなー，やってもせいぜい音響カプラという装置を持ち歩くぐらい
ごく初期のラップトップPC or ハンドヘルドPC
のせる
受話器
音響カプラー
電話器
だいたいパソコンがマトモに持ち運べないんだから実用的じゃない
15年前
10年前
やっとラップトップPCになるが重くてかさばる
TEL
1200〜2400bps
モデム
シリアル接続
プラズマディスプレイ
3年前
カラーDSTN液晶
ケータイとケーブルでコネクト
内蔵モデム
携帯電話普及時。ようやく本来の「モバイル」が実現した!!
現在
携帯機能も内蔵のPCに！もう十分!?
カラーワイド液晶
5年前
モノクロ液晶
インターネット普及時。ISDN公衆電話も増えて，かなり実現的モバイルが可能になってきた
内蔵モデム
or
TEL
PCカードモデム
TEL
1年前
カラーTFT液晶
携帯がモバイル専用のPCカード型で使える！
差込む
カード型携帯モジュール
ただ，ここにきて昨年からはじまったiモードがモバイルPC最大のライバルとなった!! iモードのユーザーはインターネットのユーザーでもあるわけだからあなどれない！
e-mail
携帯電話ネットワーク
インターネット（iモードセンタ）
HTML（CHTML）
iモード電話器
でも，まだテキストのメールとWebページ，それもiモード専用ページが中心ですよね？モバイルコンピューティングってほどじゃあ…
何をいうううう iアプリを！プチアプリを！知らんのかあああ
きゃ〜っ
ドドドド
つーかそもそも何でワシがそーゆーの知っとるかァァァ



※こういう方式の世界標準にWAP（Wireless Application Protocol）があります



IMT-2000とは何か!?

International Mobile Telecommunications2000

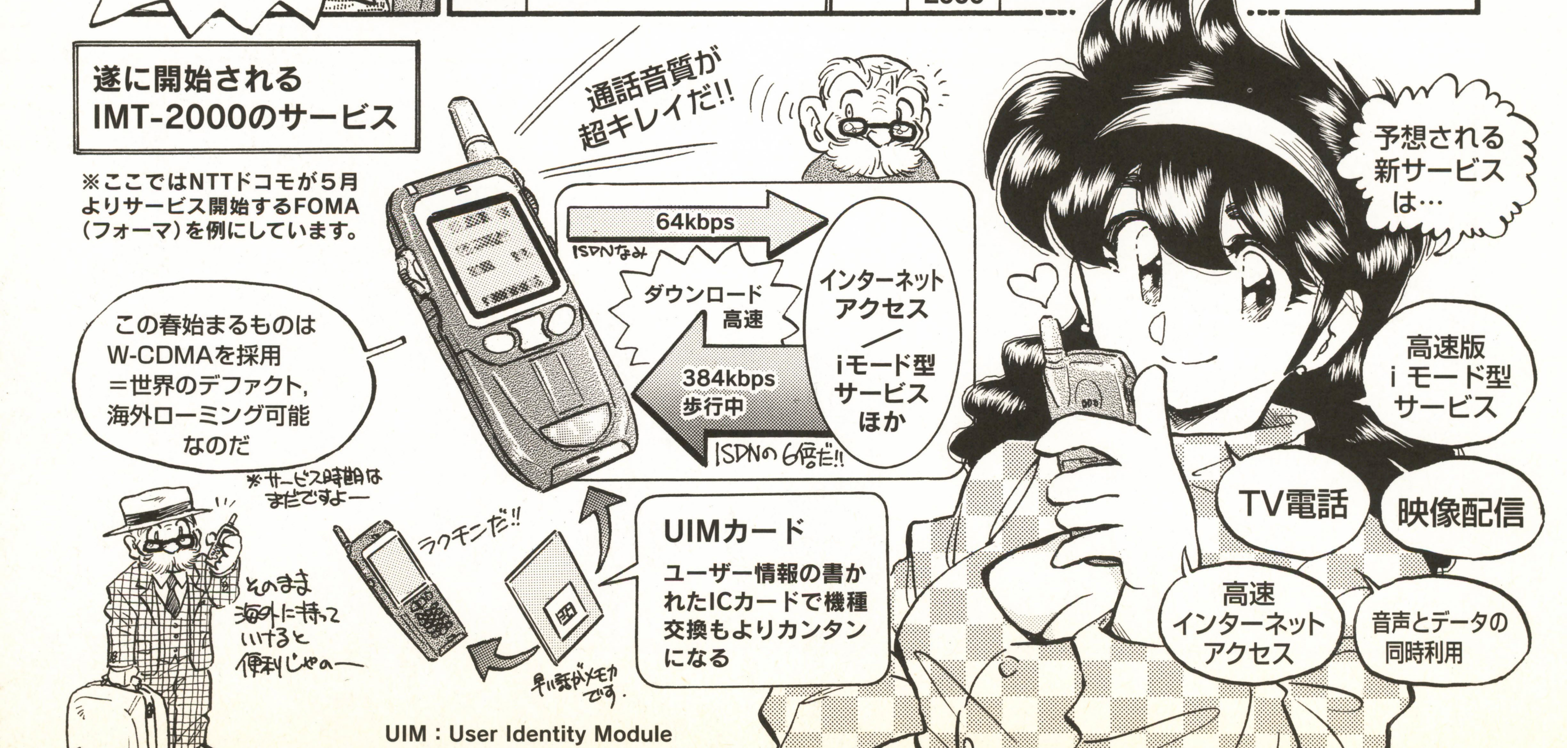
アナログ（第１世代），デジタル（第２世代）につづく，第３世代の移動通信システム。世界標準規格として決められているが，実際にはW-CDMA, cdma2000等の五つの実現方式がある。とくにデータ通信を中心とした規格で，歩行中で384kbps，屋内で静止状態なら2Mbpsの通信速度を実現できる。

現行方式（第２世代）	
PDC	日本独自の方式。現在のｉモードまではこの方式
GSM	欧州中心で，ローミングが得意。SIMカードで利用者識別も可能だ。
cdma One	北米中心，日本にもあり。音が良くて切れにくい。

現行方式（第２世代）		
IMT-2000	W-CDMA	DS（Direct Spread）-CDMA。通称日欧方式。世界で最初にサービスがはじまろうとしている。
	CDMA2000	MC（Mnlti-Carrier）-CDMA。米国方式とも呼ぶ。現行のcdmaOneからの発展型といえる。

遂に開始される IMT-2000のサービス

※ここではNTTドコモが５月よりサービス開始するFOMA（フォーマ）を例にしています。



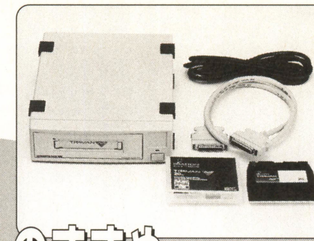
UIM：User Identity Module


IMT-2000では
端末スタイルも進化する!
ワシ的には
やっぱPDA型が
スキなんじゃが…
ベーシック端末
基本的に今まで
と同じだが
より大画面化
等高性能に
PDA型端末
TV電話，映像配信
高速インターネット等
IMT-2000の
ダイゴ味はこの
タイプで!?
携帯進化
PDA進化
従来型
携帯端末
見かけはいろいろ
でもあまり変わらない
モバイルコンピューティング進化
データ通信
専用型
ノートパソコンのPCスロット
に差し込める，データ通信
専用型の端末だ
ちなみに…
世界初!!
関東一部
2001年5月
関東，東海，関西
2001年12月
ほぼ全国で
サービス実現
（一部除く）
2002年4月
各社のIMT-2000
サービスが出そろ
って，選べる日が
まちどおしいわね
IMT-2000サービス
開始予定時期
AU（KDDI） cdma2000
→2001年秋～
J-Phone W-CDMA
→2002年6月～
気になるサービス
展開状況じゃが
1年で全国じゃよ
※ここで示されているサービス開始時期は
NTTドコモのFOMAの例です。よりくわしい
サービス時期や地域はドコモにお問い合わせください
おお──し
記事作成完了
～～～!!
かっ
影君，キミ
サボっておった
のでは……
…!?
あ──？
あ，やだなァ
ケータイの電源
切れてましたか
やるときゃヤル
男っスよォ
集中する
ために
電源も
したし…
わ～～っ
何だ何だこの
メールの山は～!?
プップップップ
ピッ
T, ToBe
Continued
なのじゃい!!
Assisted by Eiji. Special thanks to Mak.

特別企画

10ギガ以上の大容量データをバックアップする!

「テープストレージ」のすすめ



その2 ドライブ構造を知る!

文/タンベルグデータ・大島啓孝

前号では，テープストレージをどのように使うか，大量のデータの保存にはどれほど便利なものかをお話ししました。今回は，テープドライブ（テープ装置）そのものの技術的なお話をしましょう。

基本は磁石のNとS

磁気記録の基本は磁石のN極とS極で，これはハードディスクでもテープドライブでも，また電車のキップやクレジットカード，銀行のキャッシュカードでも同じです。

データを書き込む場合は，データビットが“1”の場合は，NからSまたはSからNへ磁化の状態を「反転」する，データビットが“0”の場合は磁化の状態を変化させない，という決まりにしたがって記録します。読み出しは，上のように記録された場所をコイルが通過すると電流が発生する特性（フレミングの法則）を使って“1”と“0”を判断します。

実際には，ここで説明した“1”と“0”はデータそのものではなく，データの「かたまり」を決められた規則にしたがって磁気記録に適した「記録用データ（信号）のかたまりに変換（エンコード）して記録します。そうしないと“0”ばかりが続くような場合には何も信号がない状態が続いて，しばらくあとに“1”がでてきてもタイミングが取れず「0が何個続いた後の1か」がわからなくなってしまいます。

テープドライブに適したエンコードの方式（エンコーディングルール）には，4-5GCR，1-7RLL，PRMLなどがあります。

テープドライブの種類

ここまでの話でおわかりのように，テープドライブは，「磁気テープ」という記録媒体（メディア）を磁化することで「書き込み（ライト）」を行い，テープ上の磁化状態（の変化）を検出することで「読み出し（リード）」を行います。リード／ライトは「磁気ヘッド」によって行われますが，データは連続していますのでヘッドはテープ上を移動する，またはテープがヘッド上を移動する必要があります。

ちょっと難しい話になりますが，コイルは磁化の状態の「変化」しか検出できませんので，「移動」しないと読み出しできません。

テープドライブは一般的に，テープVSヘッドの「移動」のさせ方によって大きく2種類にわけられます。ヘッドは固定されていて，テープを動かす（巻く）オーディオカセットのような「固定ヘッド方式」**（第1図）**と，高速で回転する（ドラム）ヘッドにテープをそわせ，テープはジワジワ動いていくビデオテープのような「回転ヘッド方式」**（第2図）**の二つです。

固定ヘッド方式は，記録がテープの長手方向に直線的になる**（第3図）**ため

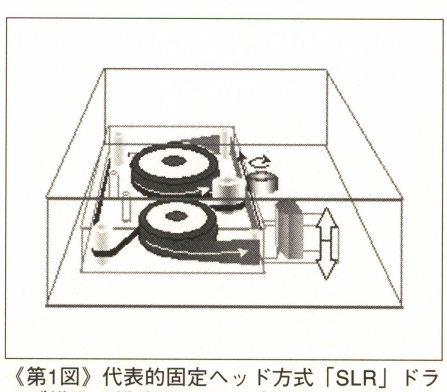
> **用語解説：テープストレージ**
>
> 磁気テープにデータを記録したり，読み出したりする装置のこと。
>
> 大きなコンピュータ・システムになると，データのバックアップに磁気テープを使うことがあります。カセットテープやビデオテープも磁気テープの一種ですが，コンピュータのデータ保存には，専用の磁気テープが使われています。

「リニアレコーディング方式」とも呼ばれます。回転ヘッド方式は，テープ上をヘッドが斜め（らせん状）にこする**（第4図）**ため「ヘリカルスキャン方式」とも呼ばれています。

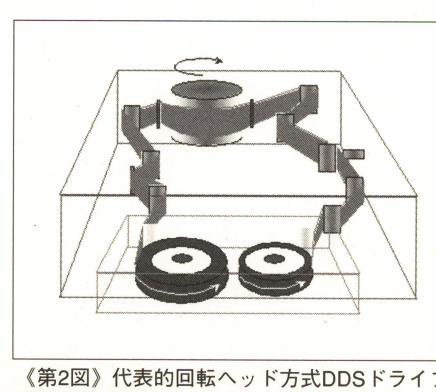
データ用として代表的な固定ヘッド方式は，DLT（Dig，ital Linear Tape）トラバンNS，SLR（Scalable Linear Recording），LTO（Linear Tape Open）など，回転ヘッド方式はデータ8mm，DAT（Digital Audio Tape，DDS: Digital Data Storage），AIT（Advanced Intelligence Tape）などです。

歴史的には，固定ヘッド方式が古くからありましたが，ビデオやデジタルサウンドのような高い周波数帯の信号を記録／再生するにはテープを高速で走らせる必要があるため，不向きでした。

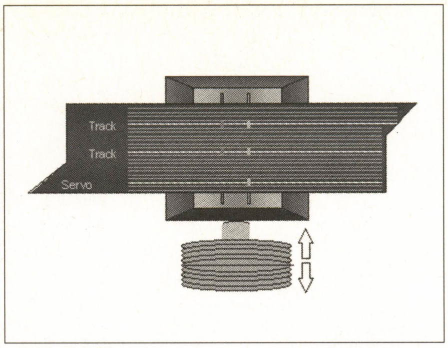
そこで，「テープを速く回せないなら，



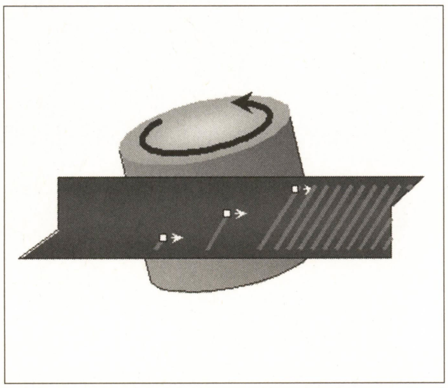
《第1図》代表的固定ヘッド方式「SLR」ドライブ構造の模式図：テープはカートリッジに入ったままで構造はシンプル



《第2図》代表的回転ヘッド方式DDSドライブ構造の模式図：テープがカートリッジから引き出されドラムに巻きつけられる



《第3図》代表的固定ヘッド方式「SLR」ドライブ構造の模式図：データはテープの長手方向に書き込まれる



《第4図》代表的回転ヘッド方式DDSドライブ構造の模式図：テープに対しヘッドを回転させ斜めに（らせん状に）書き込む

ヘッドを速く動かしましょう」という発想でできたのが回転ヘッド方式です。当時，記憶容量と転送速度には目をみはるものがありました。しかし，この方式は，固定ヘッド方式と同じようにテープを巻くことに加え，ヘッドを回転させるとか，高速で回転するヘッドにテープを巻きつけるため，カートリッジからテープを引き出す機構が必要となり，構造が複雑になる問題がありました。さらに，ある程度以上のデータ速度を達成するためには，ヘッドの回転速度さらに上げるか，回転ドラムに多くのヘッドを配置する必要がある，というような技術的・経済的な問題があります。

固定ヘッド方式は，回転ヘッド方式が登場したころは容量，速度ともに劣っていましたが，構造のシンプルさから故障の少ない装置として認識されていました。その後，ヘッドやテープの性能も良くなったため，テープ速度を上げずに，記録密度を上げることによってデータ速度を上げたり，記録幅（トラック幅）を狭くすることによって，トラック数を増やしたりして容量も地道に上げてきました。近年ではトラックずれを防ぐサーボ技術を採用したり，半導体と同じような技術で複数トラックを同時に読み書きできるマルチトラックヘッドを開発したりして，回転ヘッド方式の容量・性能を上回っています。最近は，トラックサーボにレーザを使った光技術を採用した装置も登場しました。光技術と磁気記録の技術をあわせ，将来的にはテープ1巻当たり数百ギガバイトからテラバイトクラスが達成できる計画を発表しています。

テープ技術の問題点と将来性

最後にテープ技術・テープドライブの持っている問題点と将来性について，同じ磁気記録であるハードディスクなどと比較してお話ししましょう。

【テープの耐久性】

ハードディスクでは，ディスクが回転しているときは，ヘッドはディスクに接触せず，ディスク上を飛んでいますが，テープはヘッドと接触しているため磨耗します。テープの種類にもよりますが，週1回使用している場合，だいたい1年ぐらいでテープの交換をおすすめします。磨耗のぐあいは，接触する強さ（接触圧）と接触する面積，こすれるスピードによって変わってきます。テープドライブメーカでは，接触面積が小さく接触圧の低いヘッドを開発しています。

また，複数のトラックを同時に読み書きできる「マルチトラックヘッド」によって，テープ速度を上げなくても高性能が維持できるようにして，テープの磨耗を減らしています。テープメーカでもテープ表面の処理を改善し耐久性を上げています。

【容量アップ】

ハードディスクは円盤（ディスク）の両面にしか記録できませんが，テープは長くて広い記録面をもっています。しかし，ディスクの回転は安定しているのに対して，テープ走行にはブレがありますので，トラック密度はディスクほど上げられません。しかしながら，トラックサーボ技術の開発でトラック密度も上げられるようになりましたので，広い面積とあわせてディスクに追いつかれることはないでしょう。

【性能（転送レート）】

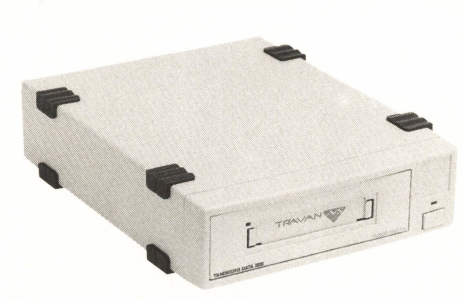
現在は，高性能のテープ装置で1秒間に10メガ以上のリード／ライトができます。前にもいいましたが，マルチトラックの採用で，理論的には1秒あたり数百メガのリード／ライトが可能です。そうなってくると，問題はインターフェースの速度のほうかもしれません。

ただし，テープドライブの転送の速さは「テープが回っていられるから速い」のです。書き込み時に十分なデータがこなかったりすると，テープの回転を止めて待っていたり，また走ったりで，実質の速度はとんでもなく遅くなってしまいます。テープドライブ側でも1～2秒程度走り続けられるデータを蓄えるバッファを持っていますが，やはり大切なことは「テープを走らせ続けられる」だけのデータを送り続けられるホスト性能とインターフェース性能です。

【価格】

テープドライブの価格は，ハードディスクやMO，CD-ROMなどほかのデータストレージに対して高めです。その理由は生産台数にあると思います。「たくさん作って売れれば安くなる」のですが，「安ければたくさん使ってもらえる」のも確かでしょう。ニワトリが先か卵が先かのお話になるかもしれませんが，たくさん使ってもらいたいものですね。

しかし，それとは別にテープの価格のお話があります。ドライブを「初期投資」とすれば，蓄えられるデータ量に対してのテープ価格はとっても安いものといえるでしょう。1ギガバイト当たり200円～400円で安全に保管できるのですから，なっとくの技術ではないでしょうか。



タンベルグデータ製テープストレージ
『Tandberg NS20Pro』（最大容量20GB）
新入学特別キャンペーン価格39,800円
（内蔵タイプ本体＋データカトリッジ1本）

パソコンでロボットを操る！

レゴ道場 桐林部屋



チャレンジ!! レゴ・マインドストーム

LEGO MINDSTORMS ROBOTICS INVENTION SYSTEM

お題：「動力からの回転運動を伝える」ギヤ

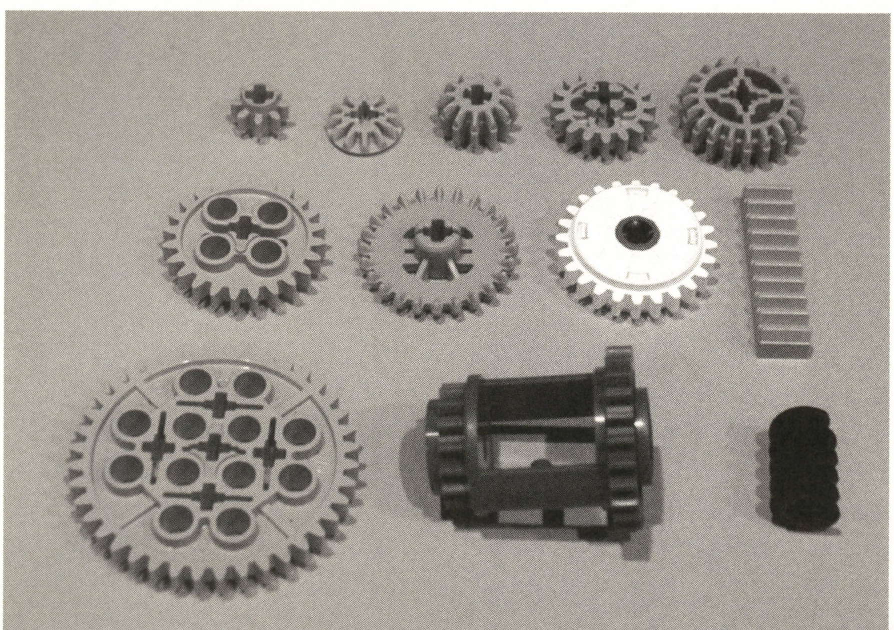


「LEGO MINDSTORMS Robotics Invention System 2.0（日本語版）」
実売価格：約23,800円
対応OS：Windows98/Me
総パーツ数：718（レゴブロック/RCX/USB IRタワー/モータ2個/タッチセンサ2個/ライトセンサ1個/日本語マニュアル/日本語ソフトウエアなど）

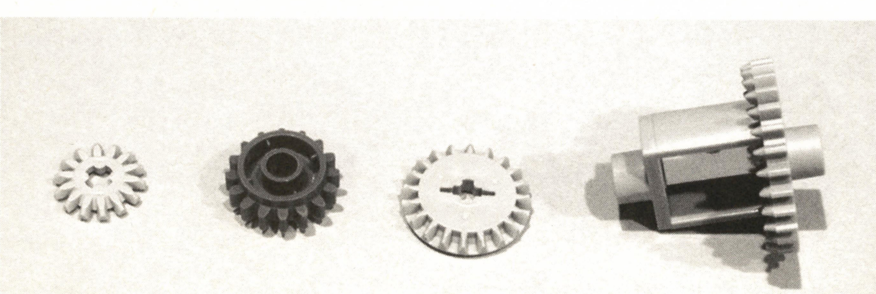
こんにちは，桐林です。今回は今まで紹介したすべてのロボットに使われているギヤについてです。たかがギヤ，されどギヤ。くわしく知ると，いろいろ面白いことがわかります。by 桐林星河

ギヤってなに?

ギヤというのは簡単に言うと，「動力からの回転運動を伝えるもの」です。しかも，ただ伝えるだけでなく，途中でスピードを落としたり，逆にスピードを上げることもできます。さらには，ギヤによってパワーもコントロールできます。それに，回転する角度も自由に変えられます。



《写真1》MINDSTORMSシリーズに入ってくるギヤ。左上から8枚歯，12枚歯，ベベルギヤ，12枚歯平ギヤ，16枚歯，20枚歯，24枚歯平ギヤ，24枚歯クラウンギヤ，24枚歯クラッチギヤ，ラックギヤ，40枚歯，デフギヤ，ウォームギヤ



《写真2》少し手に入りにくくなってしまいましたが，こんなギヤもある。左から，14枚歯ベベルギヤ，16枚歯平ギヤの軸が丸，20枚歯クラウンギヤ，28枚歯のデフギヤ

ギヤの種類

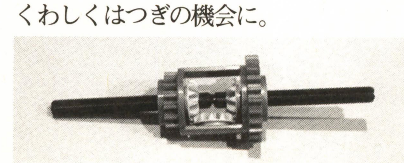
いま簡単に手に入るレゴのギヤは，平ギヤが18枚歯，12枚歯，16枚歯，20枚歯，24枚歯，40枚歯です。ベベルギヤは12枚歯，クラウンギヤは24枚歯，クラッチギヤも24枚歯です。その他に，ウォームギヤ，ディファレンシャルギヤ（以後デフギヤ），ラックギヤがあります（**写真1，2**）。

平ギヤというのは，いわゆる普通のギヤです。

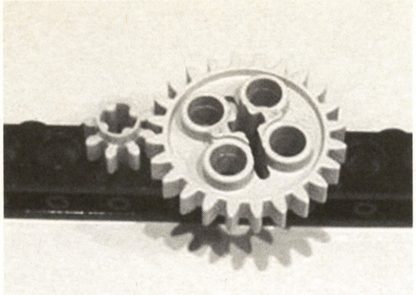
ベベルギヤというのは，傘歯ギヤとも言い，回転方向を90度変えることができます。クラウンギヤも同じ使い道ですが，形が違います。

24枚歯のクラッチギヤとは，負荷がかかったときに，うまく動力を逃がしてやる機構がついたギヤのことです。

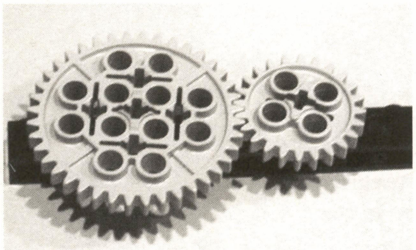
12枚歯と20枚歯の平ギヤは他のギヤと違い，変わった組みかたができます。でも今回は省略します。ウォームギヤとラックギヤも特別なギヤです（**写真3**）。くわしくはつぎの機会に。



《写真3》デフギヤはこのようにして使う。中には12枚歯のベベルギヤが入っている



《写真4》3:1のギヤ比のギヤの組みかたの一例



《写真5》このようにずれなく組める

パワーとスピード

はじめに書いたとおり，ギヤは自由にスピードやパワーが変えられます。でもたいていの場合は，パワーが上がるとスピードが下がり，スピードを上げるとパワーが下がることになります。

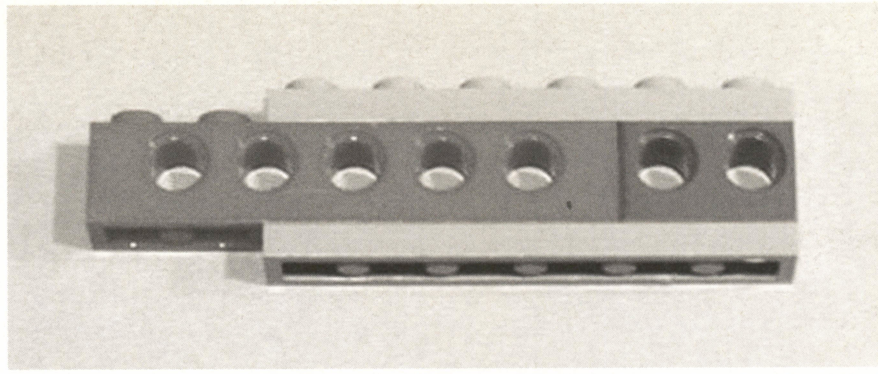
たとえば8枚歯のギヤと，24枚歯のギヤを組み合わせると，1：3というギヤ比になります（このギヤ比は，もともとのギヤの歯数を約分したものです。つまり今回の場合は8：24だったのを約分して1：3になったのです）。このときに8枚歯のほうを1回転させると24枚歯のほうは，3分の1回転します。そして，パワーは元の3倍になります。これを逆にして，24枚歯のほうを1回転させると，8枚歯のほうは3回転しますが，パワーが3分の1になってしまいます（**写真4**）。

これがパワーとスピードの関係です。パワーとスピードは比例しているのです（比例と言っても$y=ax$ではなく，$y=-ax$です。このとき，yもxも0以下にはなりません）。

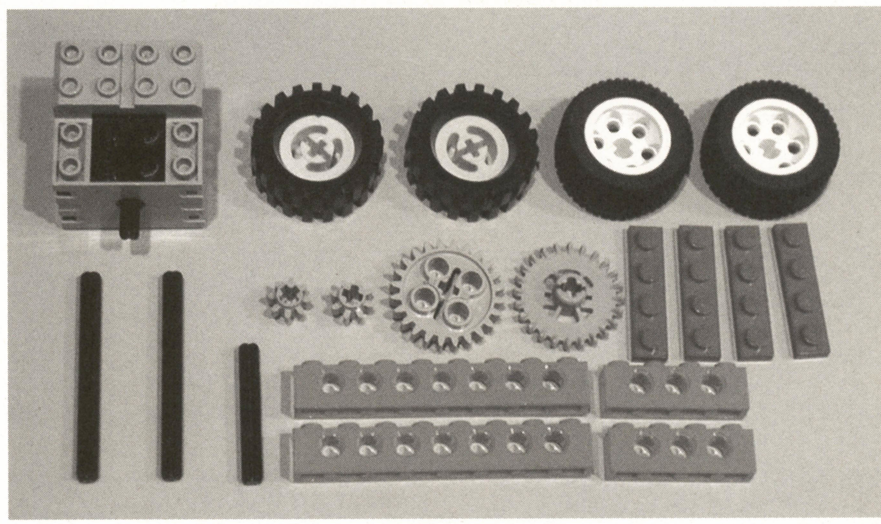
ギヤの組み合わせかた

レゴのギヤの組み合わせかたは，一見すると少なさそうですが，じつはかなりたくさんあります。

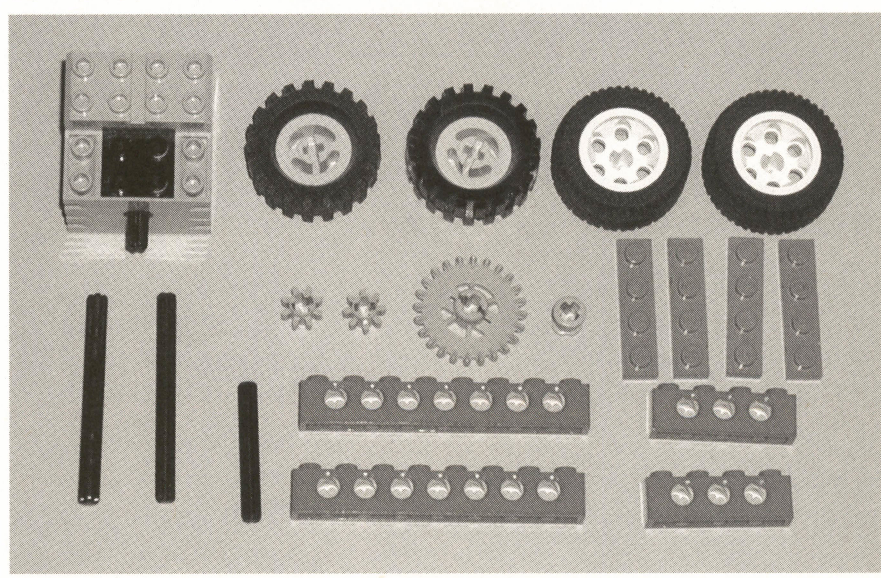
すべてを紹介しきれないので，僕が見つけた便利な方法を教えます。それは，普通にビームに組み合わせるときにしか使えないのですが，隣り合ったギヤの歯数を足して8で割ったときの商が偶数になれば，普通にギヤをかみ合わせることができますが，商が奇数になったら，ビームの穴を0.5ポッチずらさないといけないのです（例：40枚歯と24枚歯の場合　40+24=64　64÷8=8　8は偶数なので組み合わせられる）（**写真5**）。この式



《写真6》0.5ポッチずらす方法。この部品は2ポッチのビームに二つ穴があいている少ない部品。RDS，DDKには入っているがRISには入っていない

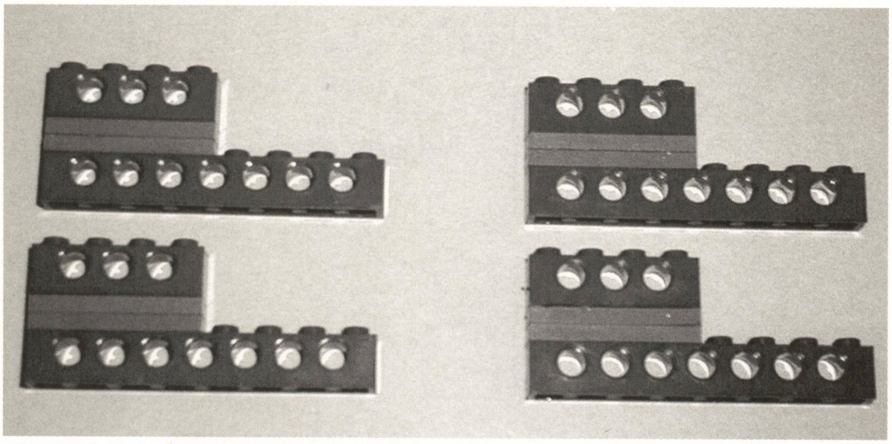


《写真7》遅い車の部品一覧。モータ1個，タイヤ4個，6のシャフト2本，4のシャフト1本，1×4のプレート4枚，1×8のビーム2本，1×4のビーム2本，8枚歯平ギヤ2個，24枚歯平ギヤ1個，24枚歯クラウンギヤ1個



《写真8》速い車の部品一覧。モータ1個，タイヤ4個，6のシャフト2本，4のシャフト1本，1×4のプレート4枚，1×8のビーム2本，1×4のビーム2本，8枚歯平ギヤ2個，24枚歯クラウンギヤ1個，ブッシュ1個

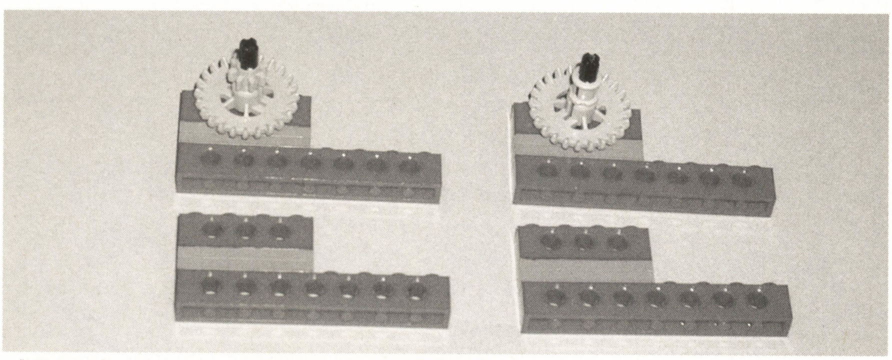
はウォームギヤとラックギヤ以外のすべてのギヤに当てはまります。覚えておくと便利です。新旧両方のデフギヤもこれに当てはまります。ちなみにビームの穴を0.5ポッチずらす場合は，**写真6**のようにします。



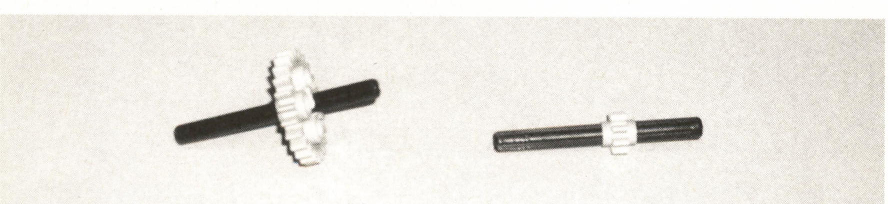
《写真9》（1）最初に写真のようにビームとプレートを組み合わせてフレームを作る（ここで紹介している写真はすべて左が遅い車で，右が速い車）



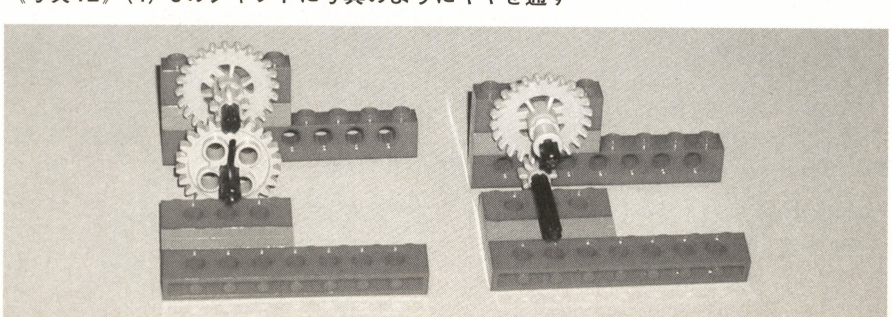
《写真10》（2）4のシャフトに写真のようにギヤとブッシュを通す（ブッシュは速い車だけ）



《写真11》（3）フレームに(2)で作った部品を差し込む



《写真12》（4）6のシャフトに写真のようにギヤを通す



《写真13》（5）フレームに(4)で作った部品を差し込む。向きに気をつけてください

実際に車を作る

さてそれでは実際にギヤを使って小さい車を作ってみましょう。もちろんRISの標準セットだけで作れます。2種類あるのでぜひ両方作って比べてみてください（**写真7〜17**）。

完成したら

完成したら走らせてみましょう。2台で比べるとスピードがぜんぜん違うのがわかります。遅い車は9：1のギヤ比で速い車は1：1です。さっき書いたスピードの関係で考えると，速い車は遅い車の9倍の速度で走っているはずです（でも実際は摩擦なんかの関係でピッタリにはなりません）。

スピードを比べたら，今度はパワーを比べてみてください。方法はタイヤを手で抑えてみるのが一番わかりやすいです（でもあんまり強く止めすぎるとモータに悪いので少しにします）。そうしたら遅い車のほうがタイヤを止めるのに力が要りますね（もしくは，止まる前に車が壊れてしまうのではないでしょうか）？これも理論上では，速い車の9倍のパワーが遅い車にはあることになります。

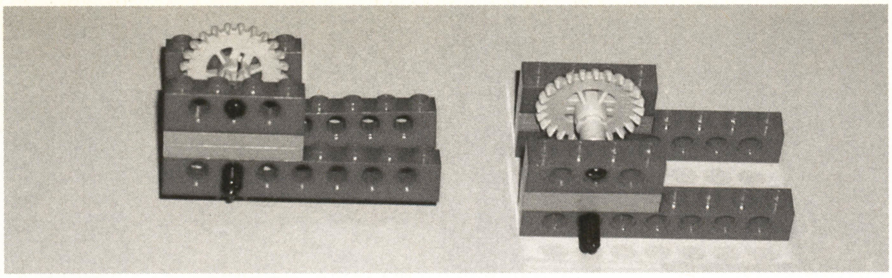
この車は，他にもギヤの組み合わせかたができます。たとえばRISのセットでは作れないのですが，**写真18**のようなものもできます。ギヤ比は約2.8：1です。自分のオリジナルの車も作ってみてください。

最後に

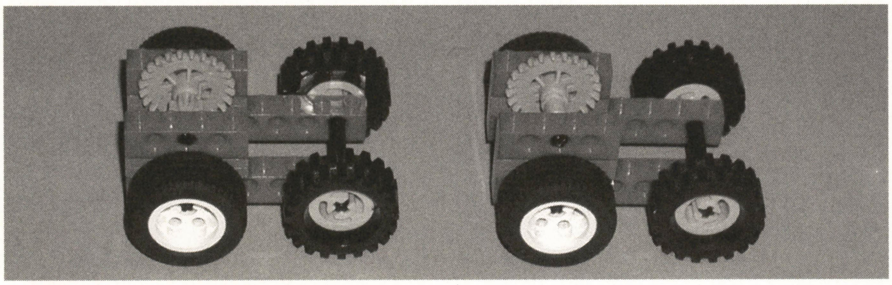
今回紹介したギヤの組みかたはほんの一部です。すべてはとても挙げ切れません。残りの組み合わせかたは自分で考え，自分で試して見つけてください。ギヤの組みかたがわかると，いろいろなロボットや車が作れるようになって，楽しくなります。

次回は

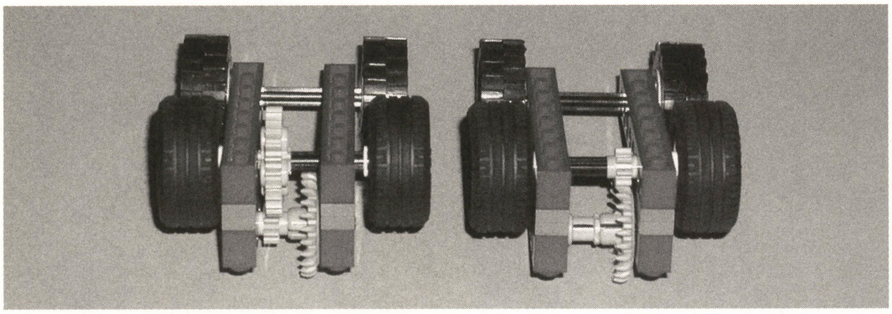
次回はビームの組みかたなど構造についてを紹介します。それでは!!



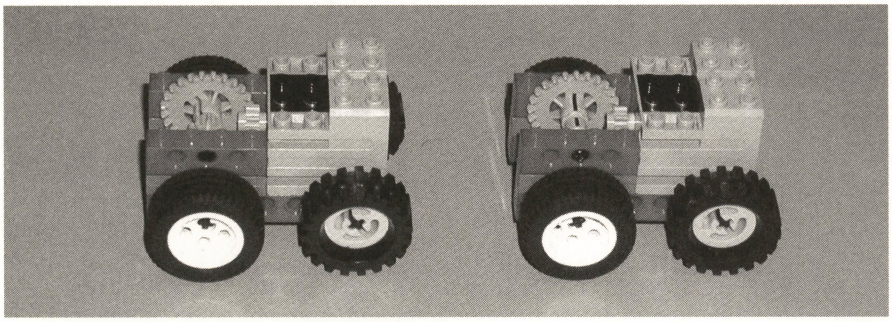
《写真14》(6) もう片方のフレームでギヤをはさみこむ



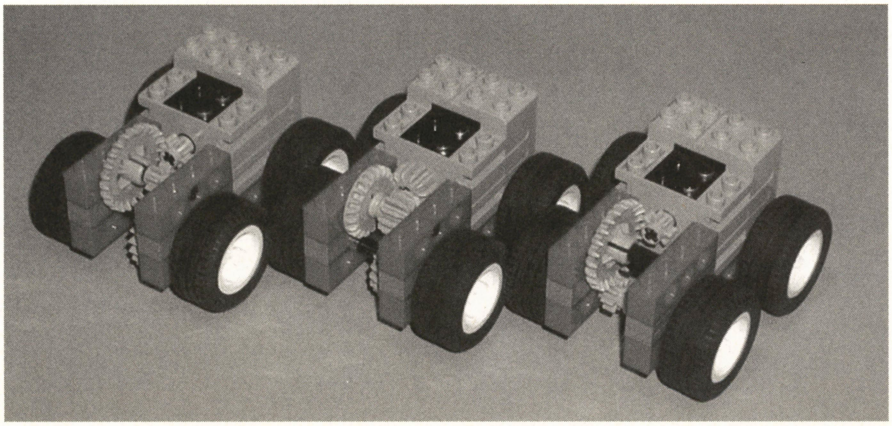
《写真15》(7) タイヤをつける



《写真16》(8) ギヤが写真のような配置になっていることを確認



《写真17》(9) モータに8枚歯平ギヤをつけて，モータをフレームにつけたら完成。長いケーブルをつけて遊んでみよう



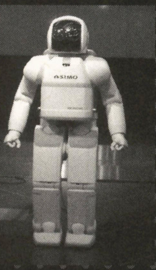
《写真18》今作ったのと同じ形の車3台。左から9:1，約2.8:1，1:1の車。見た目はほとんど同じでも速さはぜんぜん違っている

連載 第2回

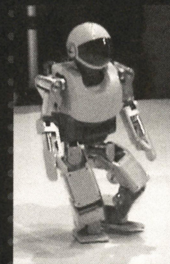
21世紀の遊びはコレ！

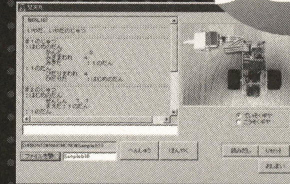
パソコンでロボットをプログラミングしよう!!

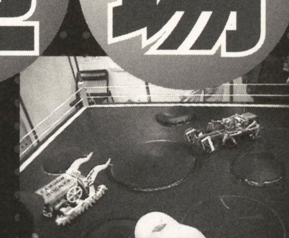
ロボ道場

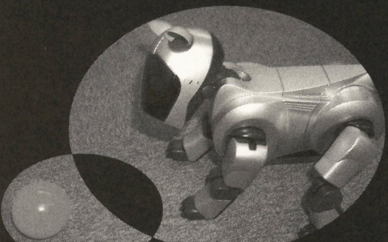












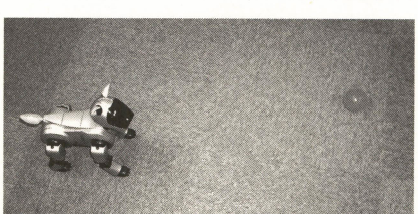
R-CODEでAIBOを操作する！②
「ボール追跡プログラム」

今月はピンクのボールを探してそれに向かっていくというプログラムに挑戦してみます。

（松原拓也）

ロボ道場 ピンク色のボールを認識する機能を使う

AIBOはピンク色のボールを認識することができるのですが，「R-CODE」にも同等の機能があります。R-CODEではボールの検出に“SEARCH”と“TRACK_HEAD”という2種類のアクションがあります。“SEARCH”はあたりを見回してボールを探すのに対して，“TRACK_HEAD”は既に見えているボールの動きを追うことができます（**写真1**）。



▲《写真1》ピンクのボールを追うプログラムに挑戦

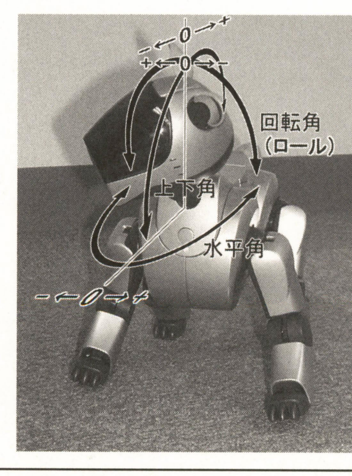
「AIBO ERS-210」	
価　格	150,000円（税別）　発売中
ソニー	
WEBサイト	http://www.aibo.com/

まず“SEARCH”を使って，以下のようなプログラムを作ってみました（**リスト1**）。AIBOの頭部には「水平角」「上下角」「回転角」という三つの自由度があります（**写真2**）。そこで頭部の水平角に合わせて，体の向きを回転させるようにしました。さらに，上下角でボールとの距離を計り，近づいたらキックしてみる機能もつけてみました（**写真3**）。

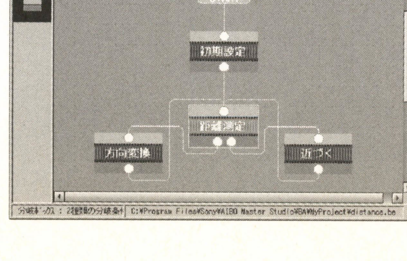
プログラムは「メモ帳（notepad.exe）」などを使って“R-CODE.R”というファイルを作成して，それを「AIBOプログラミングメモリースティック」の“OPEN-R¥APP¥PC¥AMS”フォルダに保存してください。メモリースティックにR-CODE用のシステムファイルやイン



▲「AIBOマスタースタジオ」のWEBサイト。R-CODE用のシステムファイルやインタプリタを無償でダウンロードすることができます
http://www.jp.aibo.com/ams/



◀《写真2》AIBOの頭部には三つの自由度がある



「AIBOマスタースタジオ version 1.0 日本語版 ERF-PC02」

対応機種：Windows 98/Me/2000 Professional
価格：50,000円（税別）　発売中
WEBサイト　http://www.jp.aibo.com/ams/
セット内容
・AIBO Master Studio Install CD
・AIBOプログラミングメモリースティック
・取扱説明書

＊ワイヤレスLANの環境，またはメモリースティックスロットを備えたパソコンが必要です
＊「AIBOマスタースタジオ」は最新のAIBO（ERS-210）にのみ対応しています
＊「AIBOワイヤレスLANカード ERA-201D1（別売）」を装着すれば，無線LANによる効率的なプログラミングができます。

《リスト1》「ボール追跡プログラム」"OPEN-R¥APP¥PC¥AMS¥R-CODE.R"

```
PLAY:ACTION:STAND

DO
    PLAY:ACTION:SEARCH:PINK_BALL
        //ボールを探す
    WAIT

    IF:Pink_Ball:==:1:THEN
        //ボールが見つかった

        IF:Head_Tilt:<:-45:THEN
            PLAY:ACTION:KICK //キック!
            WAIT
        ENDIF

            //首の水平角度分回転
        PLAY:ACTION:TURN:Head_Pan
        WAIT

        PLAY:ACTION:WALK:0:50  //前進
        WAIT
    ELSE
        //ボールが見つからなかった
        PLAY:ACTION:TURN:30
            //水平角度30度回転
    ENDIF
LOOP
```

《リスト2》「ボール追跡プログラム改良版」"OPEN-R¥APP¥PC¥AMS¥R-CODE.R"

```
PLAY:ACTION:STAND

DO
    PLAY:ACTION:TRACK_HEAD:PINK_BALL
            //ボールを目で追う

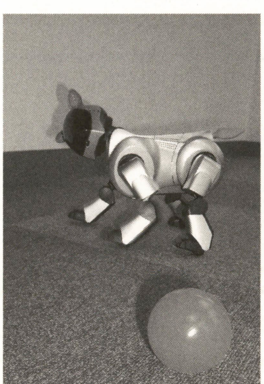
    IF:Pink_Ball:==:1:THEN
            //ボールが見つかった

        IF:Head_Tilt:<:-45:THEN
            PLAY:ACTION:KICK //キック!
            WAIT:2000
        ENDIF

            //首の水平角度分回転
         PLAY:ACTION:TURN:Head_Pan
         WAIT:2000

        PLAY:ACTION:WALK:0:50  //前進
        WAIT:1000
    ELSE
        //ボールが見つからなかった
        PLAY:ACTION:SEARCH:PINK_BALL
            //ボールを探す
        WAIT
    ENDIF
LOOP
```

タプリタが入っていない場合は，AIBOマスタースタジオのWEBサイトからダウンロードしてください（「R-CODE」のくわしい使用方法については2001年4月号の「ロボ道場」を参照してください）。



◀《写真3》上下角が下がっているため、あらぬ方向にキックを連発。水平角とANDして判定したほうがよさそう



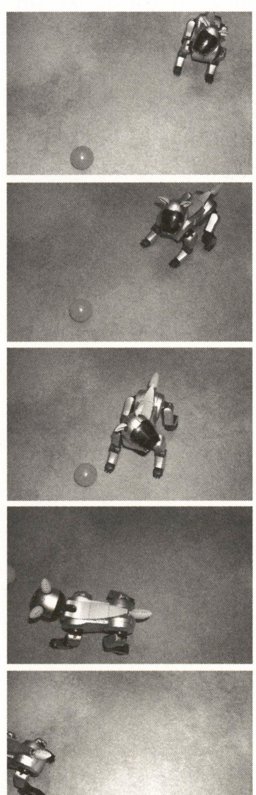
▲《写真5》"TRACK_HEAD"で"WAIT"するとボールを見つめて固まってしまう

ロボ道場 プログラム実行

では，メモリースティックをAIBOに装着して電源を入れてみます。AIBOがうまくボールに向かって歩き出しました。しかし毎度毎度，あたりをグルグルと見回しながら歩くのは，何かぎこちない感じです（**写真4**）。

そこで，プログラム中で"SEARCH"だった部分を"TRACK_HEAD"にしてみました。これなら首の動きに無駄がないはず……と思ったら今度は，ボールをにらんだまま固まってしまいました（**写真5**）。じつは"TRACK_HEAD"でボールを追跡すると"WAIT"部分で止まってしまうようです。かといって"WAIT"を取ってしまうと，高速にループしすぎてアクションが連続的に登録されてしまいます。

そこで，いろいろと試行錯誤の結果，改良してみたのが**リスト2**です。まず，"TRACK_HEAD"でボールを見つけたら"TURN"で体を回転させるのですが，"WAIT:2000"として移動にかかる時間を強制的にウエイトさせています（数値の単位はミリ秒）。"TRACK_HEAD"に失敗した場合は"SEARCH"でボールを探します。



◀《写真4》リスト1のプログラムの実行結果



ロボ道場 サッカーもできる!?

この「ボール追跡プログラム」を2台のAIBOに組み込んで，サッカーをさせてみたいのですが，果たしてうまくいくでしょうか。来月もR-CODEを使ったAIBOのプログラムを紹介します！

《第1表》R-Code System変数一覧

```
Head_Tilt  頭部上下角               [°]
Head_Pan   頭部水平角               [°]
Head_Roll  頭部回転角               [°]
Mouth      口の角度                 [°]
Tail_Pan   尻尾水平角               [°]
Tail_Tilt  尻尾上下角               [°]
RFLeg_1    右前脚第1関節角          [°]
RFLeg_2    右前脚第2関節角(肩)      [°]
RFLeg_3    右前脚第3関節角(膝)      [°]
LFLeg_1    左前脚第1関節角          [°]
LFLeg_2    左前脚第2関節角(肩)      [°]
LFLeg_3    左前脚第3関節角(膝)      [°]
RRLeg_1    右後脚第1関節角          [°]
RRLeg_2    右後脚第2関節角(肩)      [°]
RRLeg_3    右後脚第3関節角(膝)      [°]
LRLeg_1    左後脚第1関節角          [°]
LRLeg_2    左後脚第2関節角(肩)      [°]
LRLeg_3    左後脚第3関節角(膝)      [°]

Batt_Rest  バッテリー残量           [%][ 0～100%]
Batt_Temp  バッテリー温度           [℃][-20～80℃]
Body_Temp  体温                     [℃][-20～80℃]
Distance   障害物距離               [mm][ 0～900mm]
Head_ON    頭 Touch                 スイッチが押されると 1※
Head_OFF   頭 Touch                 [ms]スイッチが押されていた時間
Head_Pat   頭 Pat                   撫でられたことを検出すると 1※
Head_Hit   頭 Hit                   叩たかれたことを検出すると 1※
Head_LONG  頭 LongPat               長時間(約3秒)押し続けていると 1※
Back_ON    背 Touch                 スイッチが押されると 1※
Back_OFF   背 Touch                 [ms]スイッチが押されていた時間
Back_LONG  背 LongPat               長時間(約3秒)押し続けると 1※
Jaw_ON     顎 Touch                 スイッチが押されると 1※
Jaw_OFF    顎 Touch                 [ms]スイッチが押されていた時間
Jaw_LONG   顎 LongPat               長時間(約3秒)押し続けると 1※

RFLeg_ON   右前足 SW                右前足の肉球が押されると 1※
RFLeg_OFF  右前足 SW                [ms]スイッチが押されていた時間
LFLeg_ON   左前足 SW                左前足の肉球が押されると 1※
LFLeg_OFF  左前足 SW                [ms]左前足の肉球が押された時間
RRLeg_ON   右後足 SW                右後足の肉球が押されると 1※
RRLeg_OFF  右後足 SW                [ms]右後足の肉球が押された時間
LRLeg_ON   左後足 SW                左後足の肉球が押されると 1※
LRLeg_OFF  左後足 SW                [ms]左後足の肉球が押された時間

Pink_Ball  PINK_BALL が見えていれば 1、見えていなければ 0

Temp_Hi    高温警報（安全の為、20秒後に強制電源オフされます）
Temp_OK    温度復帰（強制電源オフされるため、検知されません）

Year       年(2000～)
Month      月(1～12)
Day        日(1～31)
Hour       時(0～23)
Min        分(0～59)
Sec        秒(0～59)※※分解能2秒
Dow        曜日(Sun(0),Mon(1)...Sat(7))

Seed       乱数の種（初期値 1）

Status     ステータス※
           0 通常起動／1 復帰回復
           転倒復帰した場合、プログラムは先頭から再実行されます。
           その時に 1 にセットされます。

ontext     コンテクスト値

ait        動作完了待ち数

lock       クロック(32ms毎に+1される)

Trace      トレースレベル（専用トレーサー用に予約済）
Debug      デバッグレベル（専用デバッガー用に予約済）
AutoReset  ※印付きの1にセットされるセンサー変数を参照の度に0にリ
           セットするフラグです。
           0以外に設定すると自動リセット機能が働きます。

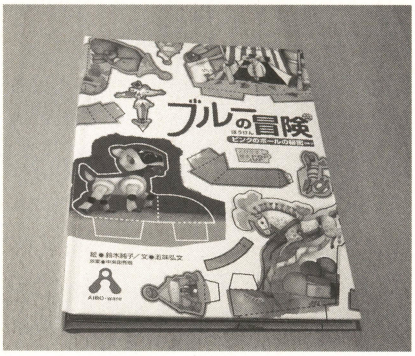
IntClock   内部クロック
           可能な限り速い速度でインクリメントされる変数。
           処理の重さによって速度は変化し、一定間隔ではない点に注意
           してください。

※これらの値は一度セットされると0には戻りません。
  SETを使って0に戻す必要があります。
```

サードパーティ製AIBO-wareが新登場

ソニー以外の企業から2本のAIBO-ware（ソフトウエア）が発売されました。

『ブルーの冒険』 AIBOが絵本を読んでくれる!



「ブルーの冒険　～ピンクボールの秘密」（ERS-210用）
発売元：ネクストファウンデーション
販売：ソニー　価格：12,900円（税別）　発売中

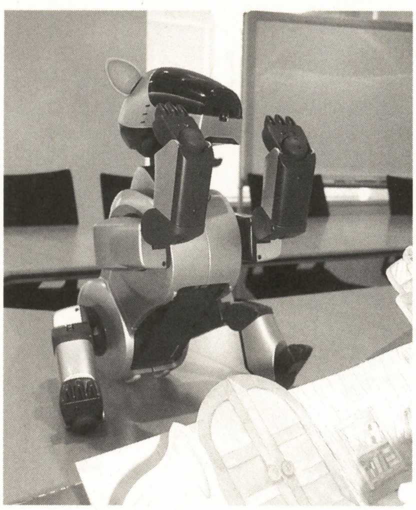
なんと，AIBOがクリアなサンプリング音声（男の子の声）で，絵本の内容を読んでくれます。

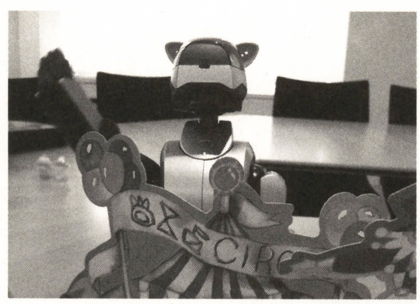
「ブルーの冒険」には全6ページの「飛び出す絵本」が付属します。絵本を開くと中から絵が飛び出してくる仕掛けになっています。

絵本を読む合図はAIBOの背中センサーを押すことで始まります。開いている絵本のページが終わったら，ページをめくってあげて再びタッチセンサを押してあげます。

AIBOの動作にはおとなしく動きの少ない「朗読モード」と身振り手振りの激しい「ダイナミックモード」の2種類が選択できます。絵本の読み上げでは，AIBOの口の動きと内容がキチンと合うように調整されています。さらに，喋りながら首や耳が自然にふるまう（微妙にフラフラと動く）ように工夫されています。

そのほか，AIBOに内蔵されている時計機能を使って，動作時間によって話の内容が何とおりかに変化します（たとえば夜では「眠い」と喋ったりします）。





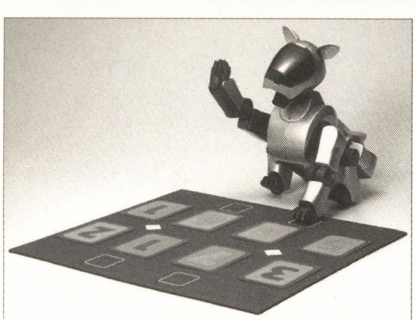
▲サンプリングされた音声で人間の言葉を話す（口も動く）AIBO

◀「ダイナミックモード」では話に合わせてジェスチャーのように手を動かしながら話します



◀「ブルーの冒険」を開発したネクストファウンデーションの中来田さん

「AIBOとあそぼ カードゲーム編」 AIBOと一緒にカードゲームを!

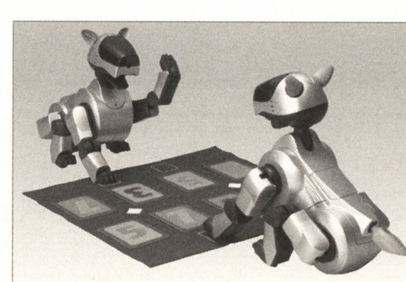


「AIBOとあそぼ カードゲーム編」
（ERS-210用，ERS-110/111用も発売予定）
企画・開発：オーパス　発売元：イマジニア
販売：ソニー　価格：18,000円（税別）
発売中

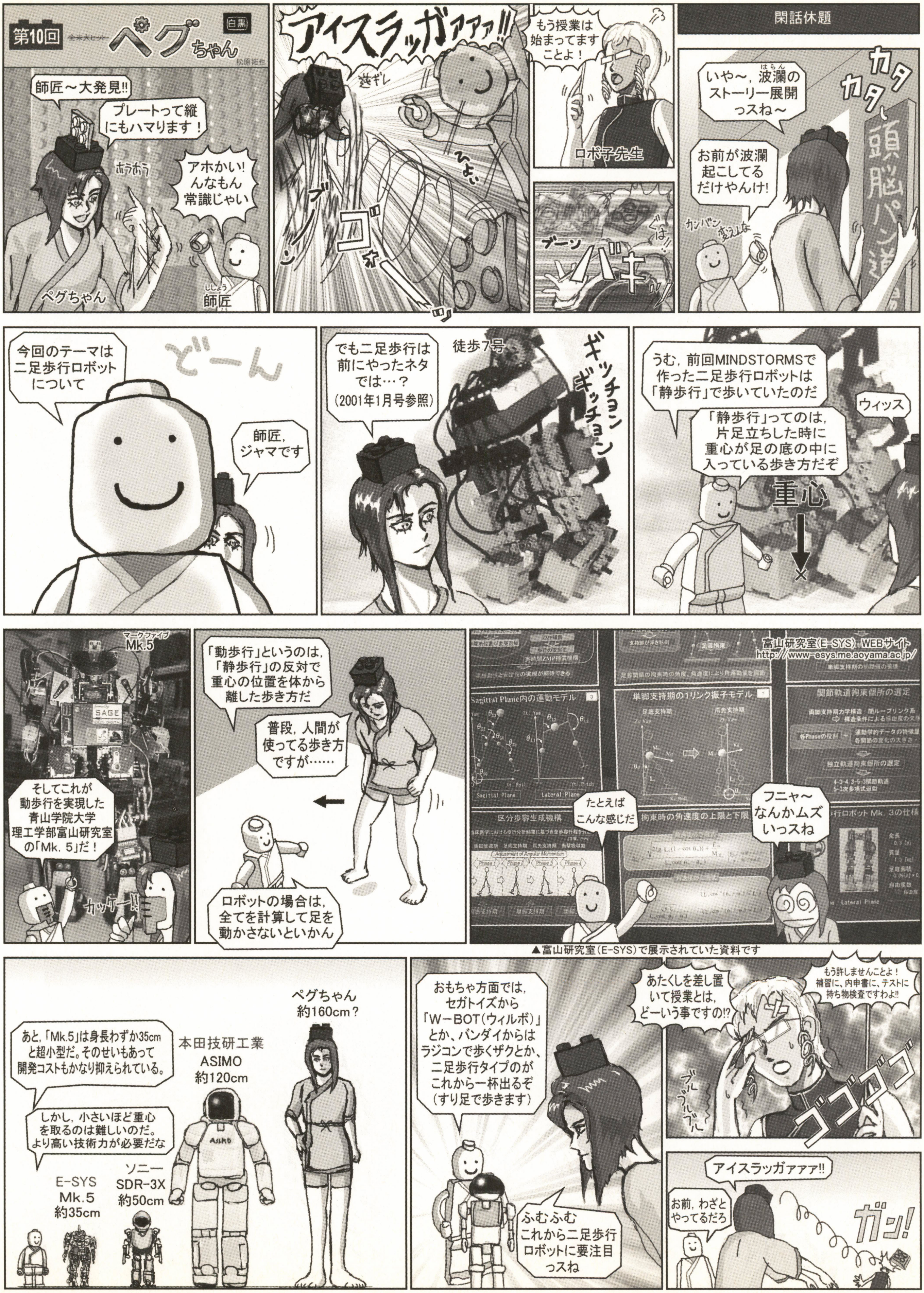
なんとAIBOとカード遊びができてしまうというAIBO-wareです。製品には特製カードと特製マットが付属し，「ポーカー」「神経衰弱」「ブラックジャック」「足し算勝負」「風林火山」の5種類のカードゲームが遊べます。

カードは7種類に塗り分けられていて，AIBOはその色と数字をもとに認識することができます。ただ，AIBOはカードを取ることはできませんので，ゲーム中にカードを並べたり，取ったりするときは人間のサポートが必要です。AIBOは，ゲームに勝つと喜んだり，負けるとくやしがったりします。そして，くりかえし遊ぶことによって，AIBOが成長して強くなっていくそうです。AIBO同士の対戦プレイも可能です。

ロボットとカード遊びをする時代が来てしまった，という感じで驚きです。ちょっと試してみたいですね。



▲人間対AIBOだけでなく，AIBO対AIBOのカードゲームも可能

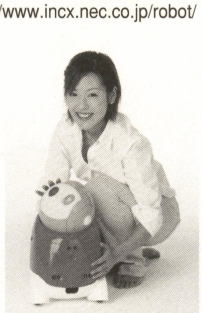


▲富山研究室(E-SYS)で展示されていた資料です

その他の気になるロボット

パーソナルロボット「PaPeRo」

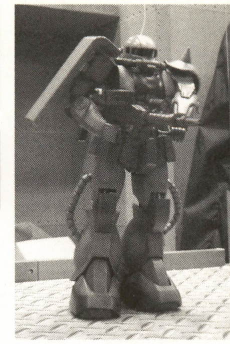
WEBサイト　http://www.incx.nec.co.jp/robot/



NECが開発した「PaPeRo（パペロ）」は高さ385mm，重さ5kgで，先の「R100」より小型化されています。PaPeRoには「お散歩モード」と「お話モード」があり，お話モードでは約650の語数を認識し，約3,000の言葉を喋ります。移動はタイヤで行い（最大20cm/秒），画像認識と超音波センサで障害物を回避します。発表会では会話によってPaPeRoに伝言を頼んだり，TVのチャンネルを切り替えさせるデモが行われていました。このPaPeRo，残念ながら発売日および価格は未定です。

東京おもちゃショー2001

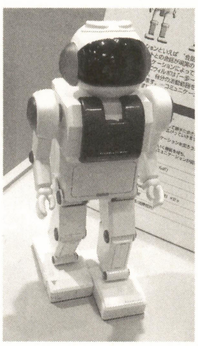
東京ビックサイトで3/22～25に行われたおもちゃの展示会にロボットが大集合しました!!



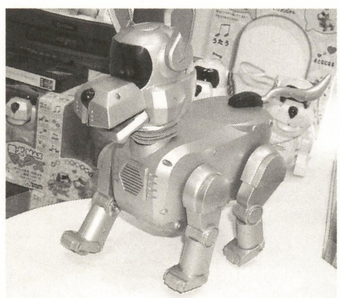
▶「MS06F-ZAKU II」／バンダイ／12月発売予定／予価98,000円／2足歩行やBB弾発射などをリモコン操作できるザク！　頭部にCCDカメラが内臓されていて，映像をリモコン上の画面に表示することができる



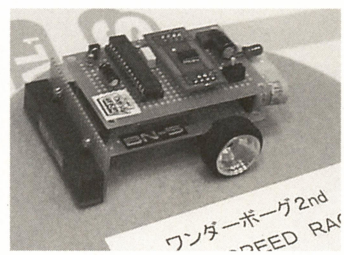
◀「コミュニケーションモンスター　ピグモン（仮）」／バンダイ／発売時期未定／価格未定／あの名キャラクターをロボット化。ピョコピョコと縦ノリで動き，目と口が可動して泣き声を発する



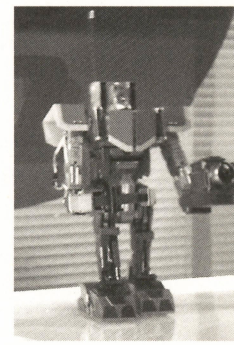
▲「W-bot（ウィルボ）」／7月発売予定／予価4,980円／二足歩行と会話によるコミュニケーションが可能なロボット。この「W-bot」の他にファミリーロボットシリーズとして「C-bot（シーボ）」や「Y-bot（ヨッチ）」などがある



▲「グレートマックス」／ツクダオリジナル／発売時期4月下旬／予価12,800円／「愛犬MAX」を大幅にグレードアップした豪華版。歩行の他に，専用リモコンに話しかけての音声認識も可能



▲「ワンダーボーグ2nd」／バンダイ／発売未定／価格未定／ワンダーボーグの後継機らしいのですが詳細は不明。試作品の状態で展示



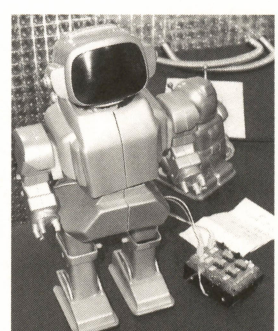
▲「DREAM FORCE 01」／タカラ／秋発売予定／予価48,000～98,000円／二足歩行の他に，腕を動かして物をつかむことも可能。ラジコンタイプの他に携帯電話（iアプリ）で操作できるタイプも発売予定



▲「IR-V」／タカラ／英語版5月発売，日本語版10月発売予定／予価8,000円／タイヤによる移動とパソコンとの通信が可能。通信可能範囲は60cm以内



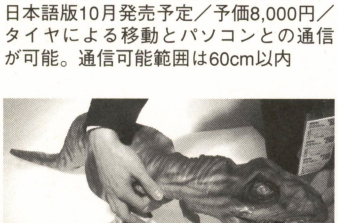
▲「BABY ROBOT-001」／トミー／10月発売予定／予価29,800円／なんと赤ちゃん型ロボット。ハイハイで歩きます



▲「RK2001」／ツクダオリジナル／7月末発売予定／予価4,980円／歩行，おじぎ，ばんざい，ダンスなどが可能なロボット。お話もする。この他にも1,980円の廉価版が7月末に，そして12,000円の完全な二足歩行ロボットが9月末に登場する予定



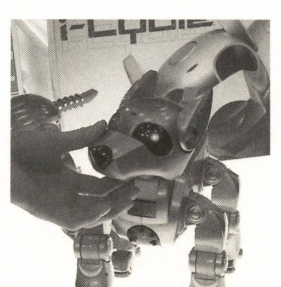
▲「DiDi（仮）」「TiTi（仮）」／コナミ／秋発売予定／価格未定／ネズミ型ロボットペット。パソコン画面に当てての通信が可能。オス型の「DiDi（ディディ）」とメス型の「TiTi（ティティ）」は仮の呼称で正式名は一般公募で決定するとのこと



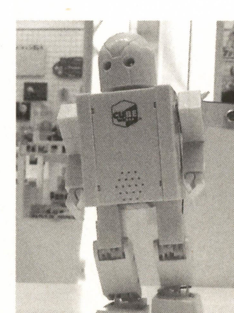
▲「恐竜を育てよう！（仮）」／トミー／11月発売予定／予価14,800円／なんと恐竜型のロボットペット。サイズが大きめで迫力がある



▲「DINKY FAMILY（ディンキーファミリー」／トミー／7月発売予定／予価3,200円／歩行型や犬型など6種類のロボットを揃えた「DINKY FAMILY」というシリーズ。顔の液晶表示で感情表現を行う



▲「i-Cybie（アイサイビー）」／トミー／発売時期未定／価格未定／犬型ロボット



▲「CAM-08」／キューブ／7月末発売予定／予価15,000円／きちんと片足を上げられる二足歩行ロボット!!

◆ 定期購読●直接購読のしおり ◆

弊社刊行の雑誌は，ご自宅直送方式の年間定期購読をお勧め申し上げます。スピーディーで確実，そしてお得な定期購読をぜひご利用下さいませ。
また　ご希望の雑誌・書籍が，ご近所の書店にない場合は右の振替用紙にて直接ご注文下さい。

■ **年間定期購読料（消費税込料金となっております。）**

電　　気　　店	¥16,000	マイコンBASICマガジン	¥6,120
電波新聞縮刷版（CD-ROM版）	¥96,000		

■ **年間購読の特典**

①特別号も通常の定価でご購読いただけます。
②電気店は13冊分の購読ができます。

◆確実な振替口座を利用してご送金下さい◆

送金手数料が不要で，確実な郵便振替でご送金をお願い申しあげます。手続きは簡単です。右の郵便振替用紙に下記の必要事項を記入し，郵便局窓口に必要金額と共にご提出いただくだけですみます。
①購読申込書に必要事項を記入ください。なお１年間，或は２年間購読かをはっきりご記入ください。
②払込通知票，払込票に送金合計金額と住所，氏名をご記入ください。払込通知票と払込票は絶対に切りはなさないようにしてください。なお，必ずフリガナをご記入下さい。
③弊社への御意見・御希望がございましたら，通信欄へご記入下さい。
④すべて記入が終了しましたら，案内文のみ切り離して郵便局窓口に金額を添えて提出してください。

∴振替によるご送金は，着本までに約２週間ほどかかりますのでお含みおきください。
お急ぎの場合は，現金書留に，右購読申込書を同封してご送金ください。

この受領証は、郵便局で機械処理をした場合は郵便振替の払込みの証拠となるものですから大切に保存してください。

ご注意

この払込書は、機械で処理しますので、口座番号及び金額を記入する際は、枠内に丁寧に記入してください。

また、下部の欄（表面及び裏面）を汚したり、本票を折り曲げたりしないでください。

（郵政省）

払込取扱票

02 東京

通常払込料金
加入者負担

口座番号（右詰めにご記入ください）
00150-3-51961

加入者名　株式会社 電波新聞社

金額　千 百 十 万 千 百 十 円 ※

料金　特殊取扱

通信欄

※定期購読誌名		購読期間	送金額（送料当社負担）
	※新規・継続	年　月号より　年間	円

御希望誌・ソフト・グッズ名	冊数	送金額（送料共）
		円
		円

フリガナ
送付先住所 □□□-□□□□　産業分類　所属又は職種

フリガナ
送付先御氏名　電話 TEL　年令 才　性別 男・女

払込人住所氏名　（郵便番号　　）※　（電話番号　－　－　）

受付局日附印

各票の※印欄は、払込人において記載してください。

裏面の注意事項をお読みください。（郵政省）（私製承認 東 第1005号）

切り取らないで郵便局にお出しください。

記載事項を訂正した場合は、その箇所に訂正印を押してください。

払込票兼受領証

口座番号　00150-3　通常払込料金加入者負担
右詰めにご記入ください　51961

加入者名　株式会社 電波新聞社

金額　千 百 十 万 千 百 十 円 ※

払込人住所氏名 ※

料金（消費税込み）　円　受付局日附印

特殊取扱

切り取って郵便局へお出し下さい。

電波新聞社出版物一覧

（消費税込料金となっております）

2000. 7. 現在

書名	価格	送料	書名	価格	送料
センサーのすべて	2,000円	〒310円	デジタル映像技術のすべて	2,000円	〒310円
MDのすべて	1,400円	〒310円	デジタル放送がわかる本	1,500円	〒310円
改訂版DVD&DVC入門基本18章	1,529円	〒310円	ビデオ技術のすべて	2,100円	〒310円
情報通信研究の最前線	2,100円	〒310円	新デジタル学入門講座回路技術編	1,800円	〒310円
TVのすべて改訂版	1,988円	〒310円	まんが電気学入門 増補改訂版	1,325円	〒310円
CD技術のすべて	2,039円	〒310円	新電気学入門講座	2,050円	〒310円
新ビデオ技術ハンドブック	2,345円	〒310円	新オシロ・ロジアナ入門講座	2,243円	〒310円
電子計測器ガイドブック	8,000円	〒580円	最新エレクトロニクス用語事典	5,300円	〒380円
Windows CE2.0がわかる本	1,300円	〒310円	SEのための情報システム工学入門	2,039円	〒310円
Word 98がわかる本	1,380円	〒310円	ハードディスク活用ガイド	1,600円	〒310円
Visual Basic 6.0/5.0講座	2,100円	〒340円	デジタル1種〈技術科目〉最新テキスト	2,100円	〒340円
F-BASIC V6.3プログラミングのすべて	2,835円	〒310円	デジタル3種受験マニュアル	3,000円	〒340円
ネットワーク対応データベース構築ガイド	2,854円	〒310円	アナログ・デジタル全資格〈法規科目〉	1,995円	〒340円
図解、コンピュータ基礎工学入門	1,937円	〒310円	特殊無線技士受験マニュアル・改訂新版	2,905円	〒340円
パソコン認定試験合格対策試験問題集3・4級	2,100円	〒310円	DOS/V製作スペシャルNo.1～7各	1,490円	〒340円

※上記以外にも新聞・出版物がございますので最寄りの本社、支局へお問い合わせ下さい。

東京本社 03-3445-8201(ダイヤルイン)
大阪本社 06-6203-3361
西部本社 092-431-7411
札幌支局 011-641-5591
仙台支局 022-227-7211
新潟支局 025-245-2526
関東総局 0273-26-3206
松本支局 0263-36-2266
名古屋支局 052-261-4541
金沢支局 076-263-8661
京都支局 075-221-8021
広島支局 082-228-5581
高松支局 0873-61-3111
鹿児島支局 099-226-3630

Q10　携帯・PHSを持っていますか？
①両方持っている ②携帯 ③PHS ④購入予定 ⑤予定なし

Q11　Q10で④の方のみ，どちらを購入する予定ですか？
①携帯 ②PHS ③両方 ④未定

Q12　使用中のプログラミング言語は？
※枠内に一つご記入ください

Q13　これから勉強したいプログラミング言語は？
※枠内に一つご記入ください

Q14　今月号の本誌で面白かった記事を三つ，下の表からお答えください

Q15　今月号の本誌でつまらなかった記事を三つ，下の表からお答えください

ご協力ありがとうございました。

■今月号の記事一覧

キリトリ線

マイコンBASIC Magazine　プログラム投稿応募カード

言語名（音源名）					
プログラム名（曲名）					
氏　名		ペンネーム		年齢	歳
住　所	〒				
TEL	※連絡可能な時間帯			応募年月日	年　月　日

キリトリ線

郵便はがき

141-8715

50円切手をお貼り下さい

（受取人）

東京都品川区東五反田1−11−15

電波新聞社

月刊 マイコンBASICマガジン 編集部

5月号 行

（フリガナ）		年齢	性別
ご氏名		歳	男・女
□□□-□□□□		電話	
ご住所	都・道 府・県		

職業：小学生／中学生／高校生／高専生／短大生／大学生／専門学校生／予備校生／アルバイト／公務員／会社員／その他（　　　）

本誌を 毎月・時々・初めて 読む　　何人で読みますか（　）人

自分のホームページを　持っている・持つ予定・持っていない・予定なし

本誌のほかによく読む雑誌は？［　　　］

ベーマガお楽しみプレゼント⑤応募欄（116ページ欄外）	希望番号	品名
	番	

キリトリ線

月刊マイコンBASICマガジン⑤ 愛読者カード

★OFコーナー宛 ♥あくせすROOM宛 ■質問・感想（本誌への掲載／可・不可）

ペンネーム：

■アンケート回答記入欄■

※黒の筆記用具でわかりやすく，はっきりとご記入ください

Q1	□	Q2	□	〔メーカー名 〕				
Q3	□	Q4	□	Q5	□			
Q6	□	□	□	□	□	□	□	□
Q7	□	□	□	□	□	〔商品名 〕		
Q8	□	□	□	□	□	□	□	□
Q9	□	□	□	□	□	□	□	□
Q10	□	Q11	□					
Q12	〔 〕	Q13	〔 〕					
Q14	□	□	□	Q15	□	□	□	

キリトリ線

■アンケートにご協力ください。

※左の愛読者カード・記入欄に，以下の質問に該当する番号をご記入ください。複数回答の質問で，回答欄をすべて埋める必要はありません。

Q1　メインで使っているパソコンは？
①デスクトップ ②ノート ③持っていない

Q2　パソコンの種類は？（Q1で③以外の方）
①自作 ②ショップブランド ③メーカー製
※枠内にメーカー名も記入してください

Q3　メインで使っているOSは？（Q1で③以外の方）
①Windows 95 ②Windows 98 ③Windows Me ④Windows 2000 ⑤Windows NT ⑥Macintosh ⑦UNIX ⑧Linux ⑨その他

Q4　パソコン歴は？
①1年未満 ②1～2年 ③3～5年 ④6～10年 ⑤11年以上 ⑥なし

Q5　インターネット歴は？
①1年未満 ②1～2年 ③3～5年 ④6～10年 ⑤11年以上 ⑥なし

Q6　パソコンの主な使用目的は？（複数回答可）
①プログラミング ②ゲーム ③インターネット ④メール ⑤ミュージック ⑥学習 ⑦CG ⑧その他

Q7　これから買いたいと思うコンピュータ機器は？（複数回答可）
①パソコン本体 ②プリンタ ③スキャナ ④デジタルカメラ ⑤その他
※わかれば，商品名も記入してください

Q8　持っているゲーム機は？（複数回答可）
①プレイステーション（PSone含） ②プレイステーション2 ③ドリームキャスト ④ニンテンドウ64 ⑤ゲームボーイ ⑥ゲームボーイカラー ⑦ゲームボーイアドバンス ⑧ワンダースワン ⑨ワンダースワンカラー ⑩その他

Q9　これから買いたいと思うゲーム機は？（複数回答可）
①プレイステーション（PSone含） ②プレイステーション2 ③ドリームキャスト ④ニンテンドウ64 ⑤ゲームキューブ ⑥Xbox ⑦ゲームボーイカラー ⑧ワンダースワンカラー ⑨ゲームボーイアドバンス

※裏面へつづく→

■メモ欄…プログラム担当者・編集部へのご意見・ご感想など

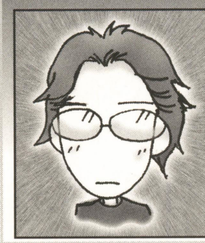
Technical News Spot（テクニカル ニュース スポット）

プログラマー必見！ Kylix（カイリックス）でLinux（リナックス）は真のデスクトップOSになる

テクスポ　2001年5月号

TEXT●石川直人

先日の3月7日，ボーランドからLinux版Delphi（デルファイ）といえる開発ツール『Kylix』の英語版が発売されました。今回は，完成前のフィールド・テスト版をもとに，Kylixをくわしく紹介していきましょう（プロスポにもどった？）。なお，今回書かれた内容は，製品版では変更されているかもしれませんので，あらかじめご了承ください。



Delphi使い。最近はWebプログラミングが増えている。家では，ノート・パソコン＋LANでインターネット接続しているのだが，最近行動半径が広がった9か月のムスコがLANケーブルで遊びたがるため，断線が頻発。早急に無線LANを導入しなければ…。

新製品

ボーランドから期待のKylixが登場！

Kylix（英語版）の動作環境は，**第1表**のようになっています。サポートされるLinuxディストリビューション[*1]は，

- Red Hat（レッドハット） 6.2以上
- Mandrake（マンドレイク） 7.2以上
- SuSE（スージィ） 7.0以上

です。

ただし日本語版では，Turbo Linux（ターボリナックス）が標準サポート・ディストリビューションとなるようです。また，とくにサポートが明記されていないディストリビューションでも，各種モジュールのバージョンさえ合えばKylixが動作するものと思われます。

Windowsアプリケーションのようなセットアップ

今回は，フィールド・テスト版の動作環境として，Turbo Linux Workstation（ターボリナックスワークステーション） 6.0aを使用しました。X-Window（エックスウィンドウ）のデスクトップ環境[*2]は，

《第1表》Kylix（英語版）の動作環境と価格

CPU	Pentium-200MHz以上
メモリ	64Mバイト以上
HDD	175Mバイト以上
ビデオ	VGA以上
その他	CD-ROMドライブ，マウスもしくは他のポインティング・デバイス
価格	サーバ版1,999ドル，デスクトップ版999ドル

“KDE（K Desktop Environment（デスクトップ エンバイロンメント））”を選択しています。Kylixのインストール前にいくつかのシステムモジュールをバージョンアップ，パッチアップ[*3]しておく必要があります。

Kylixのセットアップは，Windowsアプリケーション・ライクにできています。CD-ROMドライブのsetup.shを起動すると，Windowsアプリケーションのセットアップ・プログラムでおなじみのウィザード形式によるセットアップ作業が進められていきます（**写真1**）。

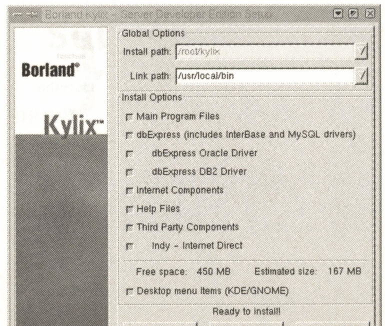
Delphiそっくりな開発環境

Kylixを起動すると，**写真2**のようにまるでDelphi（もしくはC＋＋Builder（シープラスプラスビルダー））そっくりの開発環境画面が現れます。基本的な使用感は，まったくDelphiと同じです。

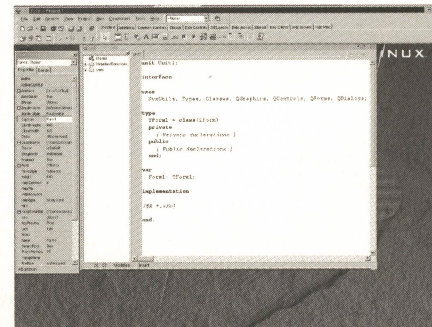
＊1　ディストリビューション
Linuxは，OSの核となる部分（Kernel）以外はいろいろなプログラムを組み合わせることができ，さまざまなメーカー／ユーザー・グループが独自の組み合わせのOSパッケージを作っている。そのようなOSパッケージのことを，ディストリビューションという。

＊2　X-Windowのデスクトップ環境
X-Windowシステムは，その外見を自由に切り替えることができるようになっている。最近では，GNOMEとKDEというデスクトップ環境が多く使われている。

＊3　パッチアップ
フィールド・テスト版では，セットアップを行う前に，glibcのパッチアップが必要だった。製品版ではどのようになるか不明。



《写真1》ウィザード形式のセットアップ・プログラムが用意されており，セットアップは簡単（でもその前の下準備は大変かも）



《写真2》LinuxのX-Window上で，まるでWindowsアプリケーションのような外観を持つKylixが起動する。Delphiそっくりだ

ただし，動作速度については，同スペックのWindowsマシンよりも落ちるでしょう。Linuxでは，裏でサーバソフトなどを動作させていることが多いので，それらのオン／オフでパフォーマンスは変わります*4。

DelphiとKylixのちがいは？

Delphiでは，Windows環境向けのコンポーネント（＝プログラム部品）として，“VCL（Visual Component Library）”というクラスライブラリ*5を採用していました。しかし，Windowsとは基本的な仕組みがちがうLinux上で，VCLをそのまま使うことはできません。

そこでKylixは，WindowsとLinux上のX-Windowの両方でうまく動作するように設計し直したCLX（Component Library for Cross platform）というクラスライブラリを採用しています。基本的な機能／構造はVCLとよく似ていますが，VCLにおけるWindowsに特化した部分を，Linuxでも動作するように汎用的に作り直したり，あるいは排除することによって互換性を高めているようです。

これは，おそらく将来発売されるWindows版Kylix*6でも使われることになるでしょう。

新しいライブラリCLXの概要

それでは，実際にKylixフィールド・テスト版のCLXと，Delphi 5日本語版のVCLをくらべてみることにしましょう。コンポーネントは，ジャンルごとに複数のタブページにわかれて収録されています。

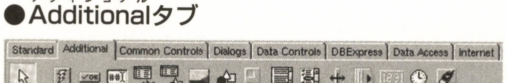
その中で，Kylixに用意されている主要なタブページの内容を紹介していきましょう。

●Standardタブ



ごくごく一般的なフォーム部品を収録したStandardタブの中身は，DelphiのVCLにあるStandardタブの中身とまったく同一です。

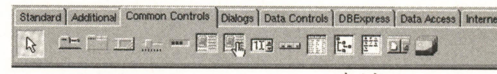
●Additionalタブ



DelphiのAdditionalタブには存在する「Static Text」と「ApplicationEvents」*7の二つのコンポーネントが，Kylixではなくなっています。これらは，どちらもWindowsに特化した仕様で作られたコンポーネントなので，Kylixでは不必要と判断されたのでしょう。

また，CLXのAdditionalタブには，DelphiではSystemタブに収録されていた「Timer」と「PaintBox」が追加されています。Kylixでは，後述するようにSystemタブはなくなってしまったので，これらはKylixではAdditionalに吸収されることになったのでしょう。

●Common Controlsタブ



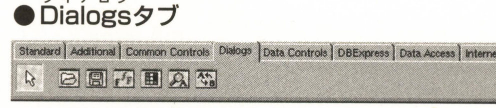
Delphiでは，SystemタブとWin32タブ*8に，Windowsの標準コントロール（DLL）をDelphiで使用するためのコンポーネント群が収録されていました。これらはWindows標準コントロールにDelphiコンポーネントの殻をかぶせたようなもので，実際の動作はWindows標準コントロールが受け持っていました。

しかし，もちろんKylixにはWindows標準コントロールは存在しないため，Delphiで行っていたように殻をかぶせる形で，Windows標準コントロール相当の機能を利用することはできません。

そこでKylixでは，DelphiでSystem／Win32タブに収録されていたコンポーネントのうち，よく使われるものと同等の機能を持つコンポーネントを，OSの機能に依存しない汎用的な形で書き直してCommon Controlsタブに収録しています。

また，SpinEdit*9やIconView*10，Text ViewerやTextBrowser*11のように，Kylixで新たに追加されたコンポーネントも用意されています。

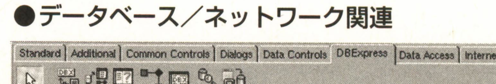
●Dialogsタブ



Windowsには標準的なダイアログ表示機能が用意されており，DelphiのDialogsタブに収録されたコンポーネントは基本的にそれらを呼びだすことでダイアログ表示を実現していました。しかし，Linuxには標準でWindows互換のダイアログはありません。

そこでKylixでは，OpenDialog，SaveDialog，FontDialog，ColorDialog，FindDialog，ReplaceDialogという主要なダイアログについて，Delphi相当の機能*12を持つコンポーネントを用意しています。

●データベース／ネットワーク関連



＊4　パフォーマンスは変わる
テストマシンは，Celeron-500MHz，メモリ128Mバイト。Workstationインストールのデフォルト状態なので，さほど多くのサーバ・プロセスは動かしていないが，それでも同スペックのWindows＋Delphi 5環境とくらべると，体感速度は非常に遅かった。

＊5　クラスライブラリ
クラスライブラリとは，オブジェクト指向プログラミング言語における関数ライブラリのこと。Delphi／Kylixでは，見た目上のプログラム部品であるコンポーネントのライブラリという意味合いが強い。

＊6　Windows版Kylix
Windows版のKylixが開発されることはほぼ確実だが，それがDelphiと統合されるのか，それともDelphiとは別の開発プラットフォームとして作られるのかは不明。

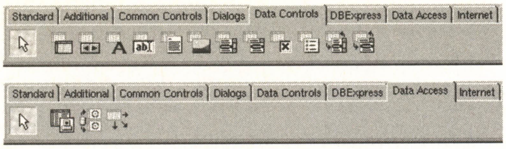
＊7　「Static Text」と「ApplicationEvents」
StaticTextは，ウィンドウハンドル（Windowsの内部でウィンドウを管理するID）を持つLabelコンポーネントとして作られたもの。ApplicationEventsは，Windowsのイベントハンドラ。

＊8　SystemタブとWin32タブ
Systemタブは，Win32以前からあるWindowsシステム関連の機能をコンポーネント化したもの。そしてWin32タブには，Windows 95以降追加された新しいWindows標準コントロールをコンポーネント化したものが収録されていた。

＊9　SpinEdit
数値変更ボタン付きエディットボックス。

＊10　IconView
ListViewのアイコン表示モードのみが独立したもので，その代わりKylixのListViewにはアイコン表示モードがなくなっている。

＊11　TextBrowser
HTMLも表示できる，高機能なテキスト閲覧コンポーネント。



データベースやネットワーク関連の機能に関しては，Kylixのパッケージの種類によって内容が大きく変わると思われます。最終的な製品版でどのような構成が採用されるかはっきりわからないため，内容について今回はくわしく触れません。

フィールド・テスト版を見る限りでは，基本的なデータベース・アクセスやネットワーク関連の機能は，Delphi相当のものがひととおり用意されているようです。Delphiにはなかった高機能なコンポーネントも用意されているようですが，それらはおそらく上位パッケージにのみ収録されるものでしょう。

Kylixでのプログラミングはこうなる！

Kylixでプログラムを組む場合，記述言語である「Object Pascal[*13]」はDelphiと同じです。また，クラスライブラリであるCLXも，外部インターフェース（プロパティ／メソッド）はほとんどVCLと同じにできています。

ですから，「コンポーネント／オブジェクト＋Object Pascal」という形でプログラムを作るぶんには，DelphiとKylixはまったく同じプログラムが動作します[*14]。

しかし，Delphiでプログラムを組む際にはWindows APIを直接呼びだしたり，Windows独自のメッセージをハンドリングしたりといったことがよく行われます。そのようなプログラムをKylixで作る場合は，別の表現で同じ機能を実現する方法を考えるか，あるいはその機能自体をあきらめる必要があります。

どのようなプログラムを作るかにもよりますが，基本的にはあまりOSに密着しない一般的なアプリケーションを作る場合は，現状のDelphiとKylixは非常に高いプログラミング上の互換性[*15]を持っているでしょう。

また，Kylixのウリとして，通常のデスクトップ・アプリケーションだけでなく，Apacheのモジュールや CGIプログラム[*16]などをネイティブ・コンパイルされたアプリケーションとして作成できる点があります。それぞれ，専用のアプリケーション・クラスが用意されています。

ホビー・プログラマーはあまり使うあてがないかもしれませんが，個人Webサイトを持っている人などは，PerlでCGIプログラムを作成する代わりに，Kylixを使用してみるのもおもしろいかもしれません。ネイティブ・コンパイルされるぶんだけ，高いパフォーマンスが期待できます。

＊12　Delphi相当の機能
おそらく，ゼロからWindowsのダイアログ相当の機能を持たせたコンポーネントを作ったのだろう。ファイル操作関連のダイアログなど，まるでWindowsのダイアログのように非常に高機能なものとなっており，使い勝手がいい。

＊13　Object Pascal
Delphi／Kylixで採用されている，オブジェクト指向対応版Pascal。プログラミング言語として，非常に美しい仕様を持っている（私感）。

＊14　同じプログラムが動作する
実際は，開発環境が自動記述するライブラリのインポートモジュール名などは異なる。

＊15　プログラミング上の互換性
dprファイル（プロジェクト・ソース）やpasファイル（各ユニットのソース）については，単なるテキスト・ファイルなので，Kylix／Delphi相互で直接読み込みができる。しかし，さすがにフォームを保存したファイル（Kylixではxfmファイル）は互換性がないようだ。まあ，コンポーネントのクラス名なども多少変わっているので，あたりまえだが。

＊16　Apacheのモジュールや CGIプログラム
メジャーなWebサーバソフトApache上で動作する，アクセス解析や掲示板プログラムなど。Perlスクリプトで書かれることが多い。

ベータ版が入手できるイベント Kylix Day開催！

Borland Kylixの発表イベント『Kylix Day』が，5月16日明治記念館で行われます。Kylixのアーキテクチャや実装されている機能の紹介，アプリケーション開発の具体的な手法の解説などが行われるそうです。また，参加者にはKylixベータ版が送られる他，Kylixキャッシュバック・クーポン5000円ぶんが提供されます。

Linuxでのアプリケーション開発に興味があるかたは，ぜひ参加してはいかがでしょうか。くわしくは，下記URLを参照してください。

■日時：2001年5月16日（水）pm1:00～pm4:30（予定）
■会場：明治記念館（東京都港区）
■定員：800名（先着順，定員になりしだい〆切）
■参加費：5,000円（税込）
■お問い合わせ：☎03-5350-9380，電子メール event@borland.co.jp
■ホームページ：http://www.borland.co.jp/kylix/kylixday/

今月の予想
2002年春モデルのパソコンは半数がLinux搭載になる！

実際にKylixを使ってみると，Windows上の洗練されたプログラミング環境をうまくLinuxのX-Window上に持ち込んでいることがわかります。

Kylixには，もっとも手軽なパッケージとして，無料の「Open Edition」が用意されることになっています。また，現状のDelphiとの高い互換性，将来的にはWindows版Kylixが発売される点などを考えると，このKylixをきっかけにLinuxのデスクトップOSとしての使い勝手が大きく向上することが予測できます。

総合的な使いやすさにおいては，まだまだWindowsのほうが圧倒的に上ですし，Kylixによってデスクトップ・アプリケーションの種類／質が大きく向上したとしても，Windowsに追いつくにはほど遠いことでしょう。しかし，それでもKylixの登場によって，少なくとも一般の人がLinux上のX-Windowを日常的に使っても，とくに困ることがないレベルまではあっという間に向上するのではないでしょうか。

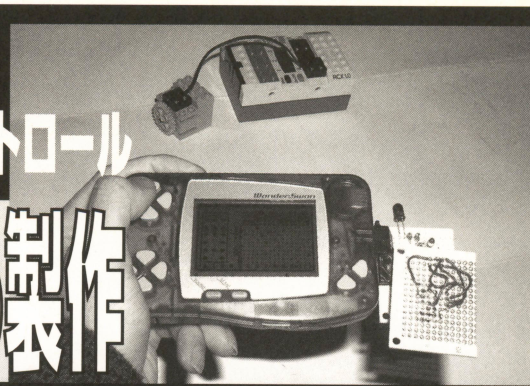
来年の今ごろには，デスクトップOSとしてLinuxを搭載したPCが，めずらしくなくなっていることでしょう。



松原拓也の自作一代記（じさくいちだいき）

連載 2回目 ワンダースワンでLEGOロボットをコントロール

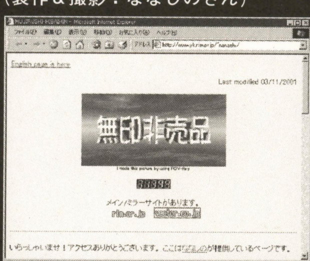
「WonderCommand-Ⅱ」の製作



先日行われたワンダーウィッチによるプログラムコンテスト「WWGP2001」では，今月の76ページで紹介しているように，ななしのさんの「WonderCommand」が見事佳作に入選されました(「WonderCommand」とは赤外線でLEGO MINDSTORMSのRCXをコントロールするソフトウエアです)。今回はその「WonderCommand」に必要な「WonderCommand-Ⅱ」というハードウエアを自作してみましょう。 (松原拓也)



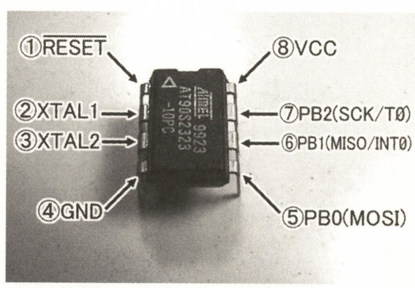
▲「WonderCommand-Ⅱ」。ワンダーウェーブの外装が使われていて非常にコンパクト（製作＆撮影：ななしのさん）



「無印非売品」
http://www.yk.rim.or.jp/~nanashi/
「WonderCommand」「WonderCommand-Ⅱ」の作者，ななしのさんのWEBサイトです

動作原理

「WonderCommand-Ⅱ」は主にAVRと赤外線LEDで構成されています。「AVR」とは8bit RISCのプロセッサを搭載したマイクロコントローラです（写真1）。AVRはプロセッサ，RAM，フラッシュROM，I/Oが1チップ化されています。I/Oは1ポートあたり10mAの電流を流せます(5V時は20mA)。赤外線LED（写真2）は発光に大量の電流が必要なので，二つのポートを合わせて電流を増やしています。ワンダースワンから9600bpsで出力されてシリアルデータは，AVRによって2400bpsに変換され，赤外線LEDの光となって出力されます（搬送波は38kHz)。

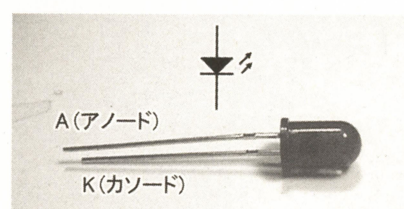


▲《写真1》ATMEL社製の「AVR AT90S2323」。8bit RISC搭載の1チップマイコン

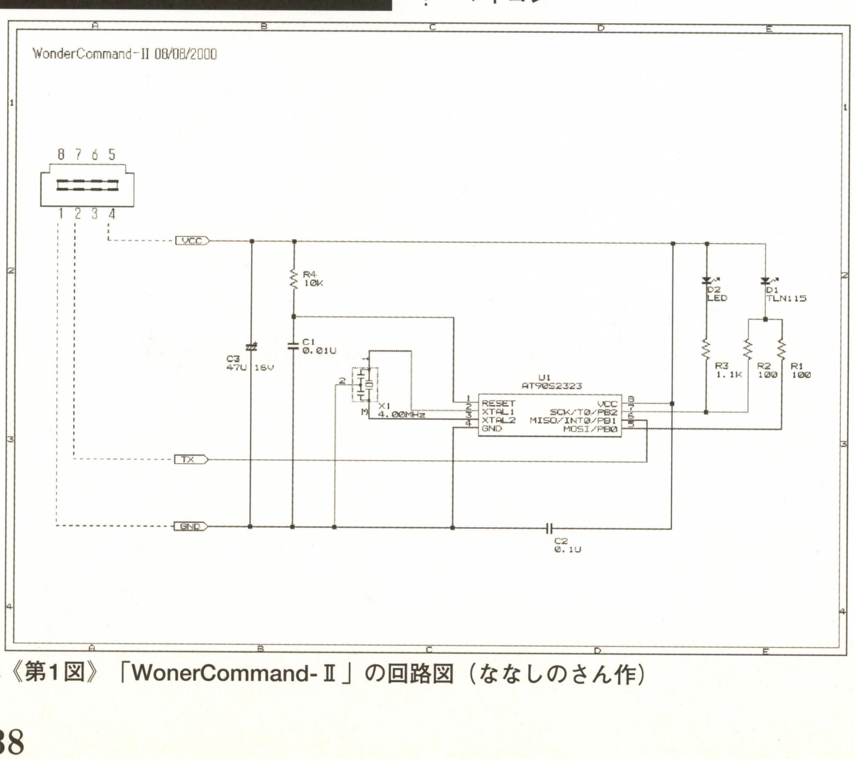
材料の購入

以下の材料を用意しましょう（写真3）。

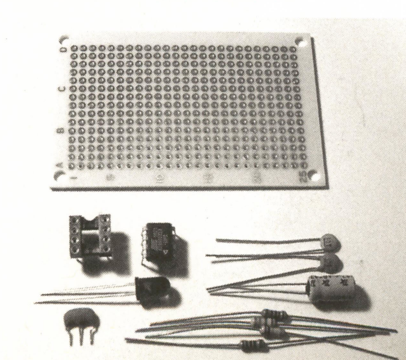
● AVR（AT90S2323または AT90LS2323。約400円）



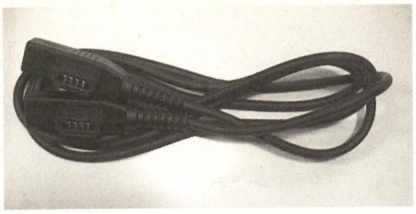
▲《写真2》赤外線LED「GL538」（シャープ製）。購入価格は40円。赤外線なので発光しても目に見えない



▲《第1図》「WonerCommand-Ⅱ」の回路図（ななしのさん作）



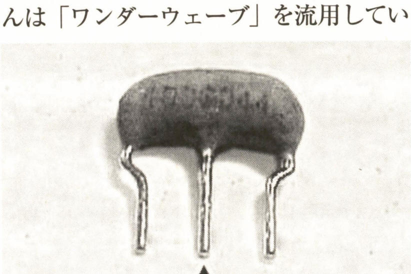
▲《写真3》購入した部品


▲《写真4》サミー製ワンダースワン用「通信ケーブル」。購入価格1,000円

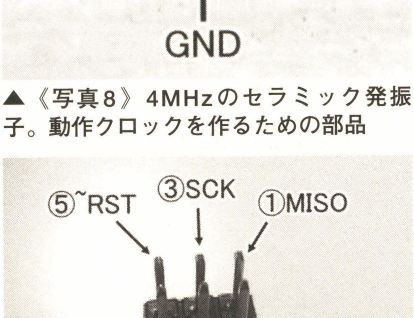
- ●セラミック発振子×1個（4MHz。約40円）
- ●ユニバーサル基板（約70円）
- ●赤外線LED ×1個（GL538やTLN115など。約40円）
- ●LED ×1個（約20円）
- ●100Ω ×2個（約5円）
- ●1.1kΩ ×1個（約5円）
- ●10kΩ ×1個（約5円）
- ●0.01uF ×1個（約10円）
- ●0.1uF ×1個（約10円）
- ●電解コンデンサ47uF ×1個（約20円）
- ●ワンダースワン外部端子用コネクタ ×1個

コネクタの分解

「ワンダースワン外部端子用コネクタ」以外の部品は，秋葉原の秋月電子通商で入手できました。問題の“コネクタ”は，単体での入手は難しいです。ななしのさんは「ワンダーウェーブ」を流用していましたが，僕は「通信ケーブル」を購入しました（写真4）。



▲《写真8》4MHzのセラミック発振子。動作クロックを作るための部品



▲《写真10》ISPアダプタ用コネクタのピン配置（6ピン）


▲《写真5》カッターナイフで通信ケーブルを分解！

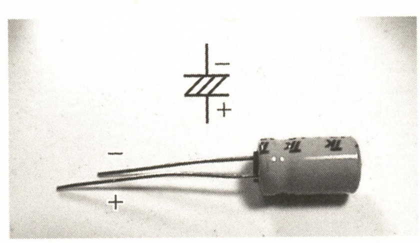
通信ケーブルは硬いゴムで固められていて，カッターで削るのに30分ほどかかってヘトヘトです（写真5，6，7）。

組み立て（その前半）

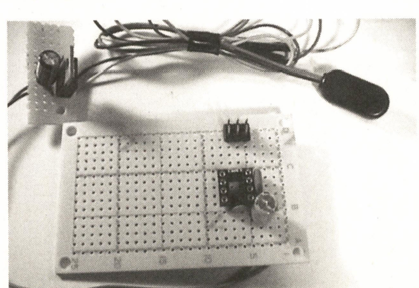
あとは回路図（第1図）どおりに，はんだ付けしていくだけです。AVRはフラッシュROMにプログラムを書き込む必要があります。今回は同じ基板上でフラッシュROMを焼いてしまおうと思いつきましたので，AVRとセラミック発振子（写真8），コンデンサ（写真9），そしてISPアダプタ用コネクタ（写真10）だけを取りつけました（写真11）。

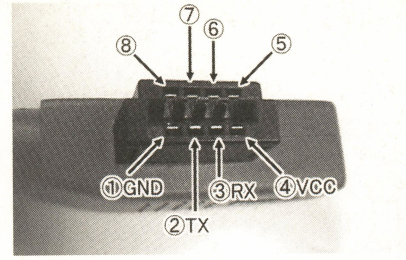
ROMを焼く！

では，ROMを焼いてみましょうか。

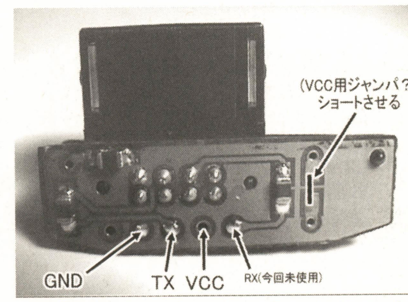


▲《写真9》電解コンデンサ。電源電圧を安定させるための部品


▲《写真11》AVR，コンデンサ，発振子，コネクタだけを取り付けた状態でフラッシュROMを焼いてみる



▲《写真6》ワンダースワン外部端子用コネクタのピン配置



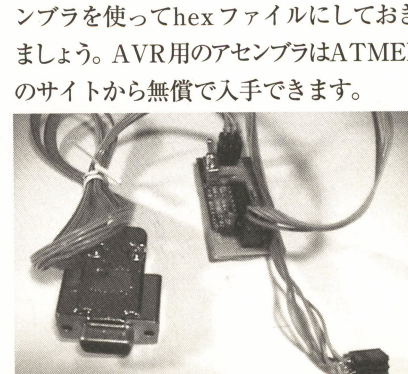
▲《写真7》コネクタのランド部分（他の製品では配置が違うかもしれません）

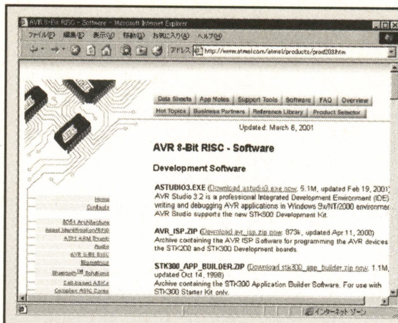
AVRシリーズには「シリアルダウンロードモード」というモードがあり，シリアル通信でフラッシュROMにプログラムを書き込むことができます（ROMの場合「焼く」という表現を使います）。

そこで，以前紹介した若松通商のキット「電光掲示板」（2000年9月号）や「MP3プレーヤ」（2000年7月号）に付属してきた「COM制御ISPアダプタ」という通信ケーブルを使ってみました（写真12）。

「電光掲示板」や「MP3プレーヤ」の原作者はChaNさんという方で，「COM制御ISPアダプタ」用の書き込みソフトはChaNさんのWEBサイトからダウンロードすることができます。回路図も公開されていて，アダプタの自作も可能です。

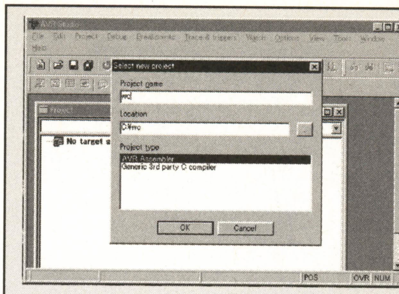
プログラムのソースはあらかじめアセンブラを使ってhexファイルにしておきましょう。AVR用のアセンブラはATMELのサイトから無償で入手できます。

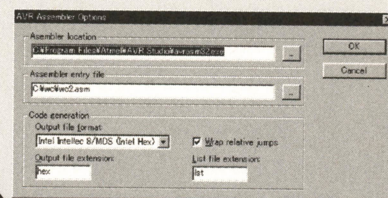

▲《写真12》シリアルポート用のISPアダプタ（AVRライタ）



①ATMEL社のWEBサイトから，AVRの統合開発ツール「AVR Studio 3.22」をダウンロードします。自己解凍ファイル“astudio3.exe”を解凍したら，“setup.exe”を実行してインストールします。

下準備としてWWGP2001のWEBサイトか，ななしのさんのWEBサイトから「WonderSCommand-II」のソースファイル“wc2.asm”（アセンブラです）をダウンロードします。そして，新しく“C:¥wc”フォルダを作って，“wc2.asm”はその中に格納します。さらに，“C:¥Program Files ¥ Atmel ¥ AVR Studio ¥ Appnotes”フォルダにある“2323def.inc”を“C:¥wc”フォルダにコピーします。

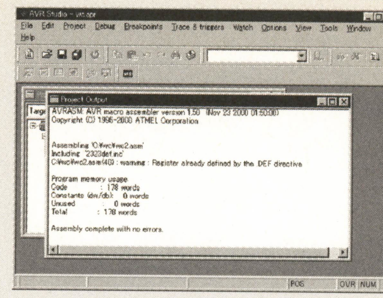




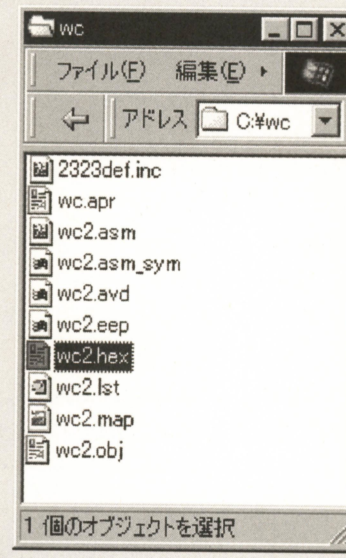
②新規にプロジェクトを作成して“c:¥wc”フォルダを登録します。

そして，“Project－Add File”をメニュー選択して“wc2.asm”を追加します。

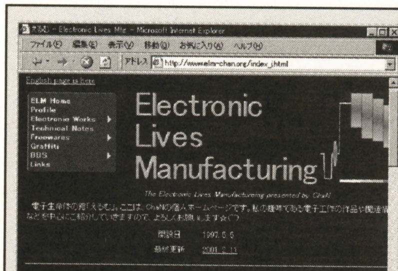
あと，“Project－Project Settings..”メニューを選択し，“Output file format”を“Intel Intellec 8/MDS (Index Hex)”に設定しておきましょう。



③“Project－Assemble”をメニュー選択して，“wc2.asm”をアセンブルします。成功すると“WC2.HEX”ができあがります。

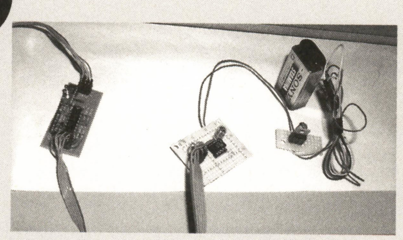


AVRのプログラム方法

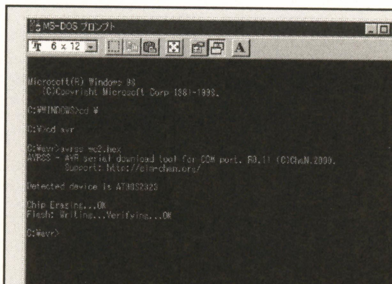


④「Electronic Lives Manufacturing」 http://elm-chan.org/
ChaNさんのWEBサイトです

ChaNさんのWEBサイトから“AVRXTOOL.ZIP”をダウンロードして，“AVRSS.EXE”を解凍します。ISPアダプタの回路図も公開されています。



⑤ISPアダプタのコネクタを基板に差し，その反対側をパソコンのシリアルポートに接続します。そして，基板に3～5V程度の電圧を加えます。電源は3端子レギュレータやワンダースワンから取り出せます。



⑥hexファイル“WC2.HEX”を“AVRSS.EXE”と同じフォルダに格納します。そして，コマンドプロンプトを起動して，

>AVRSS WC2.HEX［リターン］

と入力します。“OK”と表示されればフラッシュROMの書き込みは成功です！

組み立て（後半）

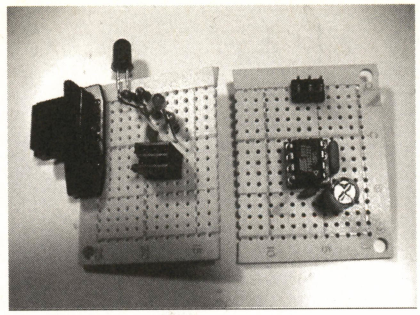
ワンダースワン外部端子用コネクタ，赤外線LED，抵抗をはんだ付けします。そして，2枚の基板を合体させます。なんか，外見がオリジナル（**写真1**）とぜんぜん違うものになってしまいましたが……とにかく，ハードウエアは完成です（**写真13**）。

完成！

まず，スワンケーブルを使って，ワンダースワン（ワンダーウィッチ）に「WonderCommand」（“wc.fx”と“wc.cfg”）を転送します。そして，ワンダースワンに「WonderCommand-Ⅱ」を差し込みます。

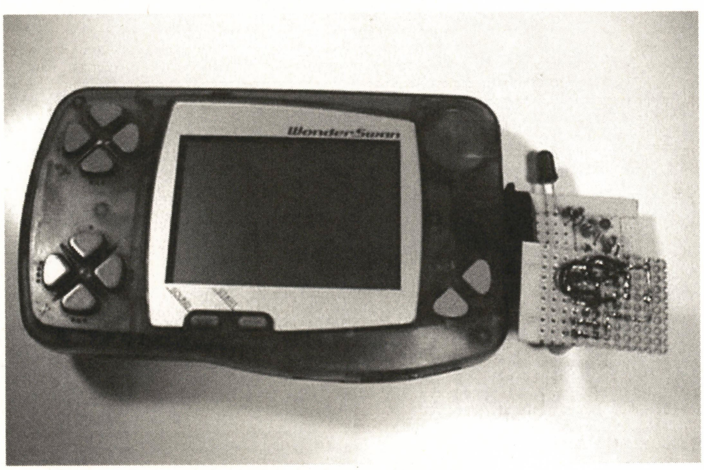
RCXの電源を入れたら，今度はワンダースワンの電源を入れて，「Wonder Command」を実行してみましょう。

ワンダースワンの方向キーを押すとモータが回りだしました（**写真14**）。見事



▲《写真13》基板を2枚に分けて，ISPケーブル用コネクタで基板同士を合体させる

に成功です!!（**写真15**）。ラジコンみたいで面白いですね。



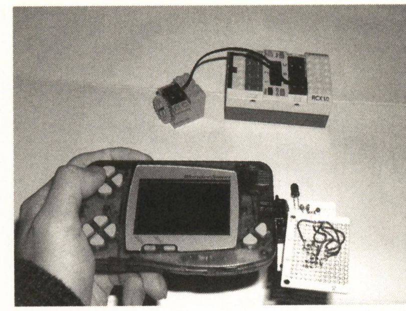
▲《写真15》「WonderCommand-Ⅱ」完成

組み立て（後半）

ワンダースワンには方向キーが二つあります。その二つの方向キーを使って，2台のロボットを同時にリモコン操作するソフトを作ってみました。まず，いしかわきょーすけさんの「MessageRouter」（2000年2月号参照）用のRCX CodeをRCXに組み込みます。

そして交互にメッセージ送信するようにWonder Witch用のプログラムを自作してみましたら，あっさり成功しました（**写真16**）。

ただ，2人で遊ぼうとなると，コントローラの奪い合いになって，ちょっと遊びにくかったです。夢のロボット対戦

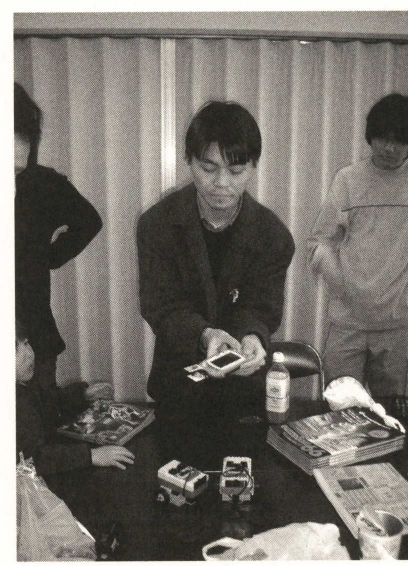


▲《写真14》「WonderCommand」で動作テスト。ボタンを押すと見事にモータが回転！

を実現するのが今後のテーマでしょうか。

しかし，AVRやPICって便利ですね。総製作費1,625円。開発期間は3日でした。

＊この記事で発生したいかなる損害も保証しませんので，自己責任で注意して自作してくださいね。



▲《写真16》「MessageRouter」のRCX Codeを使って2台のロボットを同時操作（撮影：OSAMUさん）

デジタルとアナログの境界線上で作る電子オモチャ

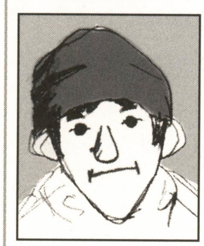
Techno テクノ Toys トイズ Laboratory ラボラトリ

今月の製作

電子楽器テミン（その2）

テクノ・オモチャ研究所

TEXT＆制作●クワクボリョウタ



デバイス・アーチスト。1998年頃から「ビット」をキーワードにメディアアート作品や電子オモチャなどを制作・発表。引越しするときになぜかパソコンのハードディスクにたまったファイルも整理してしまう。懐かしいゲームが出てきたりして全然はかどらない29歳。クワクボホームページ（http://www.vector-scan.com/）

今回は前回に引き続き，電子楽器「テミン」の製作をすすめていく。今回はプログラムの改良だ。ある条件で音を止めるミュート機能を付け，さらにパソコンのディスプレイなどの光源で音を制御できるようにプログラムをヴァージョン・アップする。そのために「割り込み」というテクニックを使うことにした。

引き続きテミン製作

前回は光センサ式電子楽器「テミン」の基本的な動作を実現した。今回はその続編として，ソフトウエアのヴァージョン・アップをしていこうと思う。前回のプログラムでは，光の強弱に合わせて音程を上下するだけの機能だったけど，せっかくワンチップマイコンを使っているんだから，もうちょっと高度な動作をさせたいところだ。そこで，明るさの度合によって音を止めるミュート機能を付け加え，さらにパソコンのディスプレイなどのブラウン管に表示される映像との連動を試みる。

ミュート機能をつける

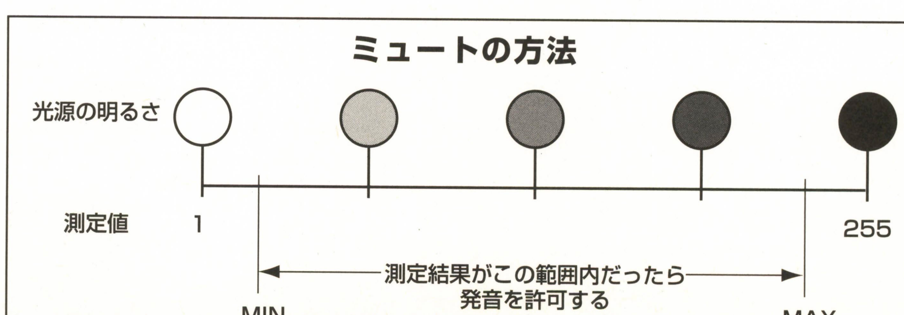
ミュート機能は，なんか音が鳴りっぱなしだとマヌケな感じがするのでつけることにした。具体的には，明るさに上限と下限を設定して，測定した明るさがその範囲外にあるときには音を鳴らさないようにする（**第1図**）。そこで，充電完了までの待機時間をカウントし，測定結果を数値として得て，発音する範囲と比較ができるようにしよう。

カウントするためには「変数」が必要なことはいうまでもない。PICはレジスタと呼ばれるデータメモリを持っていて，それぞれ1バイトの数値を読み書きすることができる。それぞれのレジスタにアクセスするにはデータメモリの番地を指定すればよい。アセンブラでは，直接番地を指定したのでは分かりにくいので，それぞれに適当な名前をつけて使うのが普通だ。また，PICはメモリ上にマップされているレジスタとは別にWレジスタと呼ばれる汎用のレジスタを持っている。これはメモリ上のレジスタを読み書きするときや演算などに使う。データシートを参照して，レジスタの使い方，命令セットを理解してほしい。

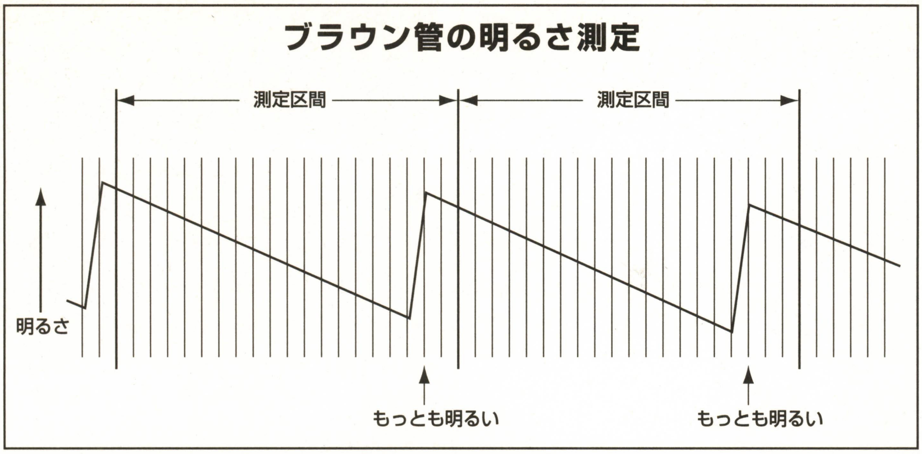
ブラウン管との連動

ブラウン管との連動について説明すると，パソコンやテレビの画面上にいろんな映像を表示して，その明るさを光センサで拾えば自動演奏みたいなことができるんじゃないかという発想だ。これって，ただ単に画面に光センサを当てればいいんじゃないかと思うでしょう？　しかし実際はそうはいかない。試してもらえば分かると思うけど，前回のプログラムのままではまともな音程感のある音は出てくれない。ジュ～ッというようなノイズに近いような音が鳴るだけだ。

これは，ブラウン管の発光の仕方に原因がある。ブラウン管は太陽光のように常に光っているわけではなくて，何十万個とあるそれぞれの画素を順番に光らせて（スキャンして）画像を表示している。そのため，見た目には気にならないけど，じつは1秒に数十回程度点滅している。CdSはそのくらいの速さの光の変化には敏感に反応するので，結果として1秒間に数十回の周期で音程が極端に上下してしまう。コレはコレで面白いんだけどね。まぁ今回はこの点滅による音程の変化をプログラムで吸収して，一定の音程を出す工夫をしよう。



《第1図》このように発音する範囲を定め，範囲外なら鳴らさないように設定するのだ



《第2図》ブラウン管の点滅を吸収する測定方法。測定区間の間に何度も測定し，もっとも明るい（充電時間の短い）値を有効な結果とする

《第1表》処理の優先度。タイミングが重要な処理と，そうでない処理を分けて考える

優先度の高い処理	優先度の低い処理
サウンド出力（圧電素子の電圧の切り替え）	明るさが発音の範囲内にあるかチェック
明るさ計測（充放電の切り替えと電圧の監視）	ブラウン管の点滅による測定値のバラツキを吸収する

やり方は簡単。ブラウン管の点滅周期よりも長い時間（以下，測定区間と呼ぶ）を設定し，その測定区間内に何度も明るさの測定を行う。そして，その中でもっとも明るい値（測定カウンタのもっとも小さな値）を最終的な測定値として，サウンドの音程にセットすればいいのだ（**第2図**）。

処理の優先度を考える

以上2点を付け加えようとすると，いろいろな問題が浮かび上がってくる。前回のような単純な動作であれば，PICは常に充電監視ピンRA4の電圧をチェックするだけでいいので，充放電が完了した瞬間にサウンド出力を切り替えることができた。しかし，新たな機能を付けるとなると，明るさの測定値をレジスタに保持したり，測定値の比較をしたり，一定の測定区間を測る必要があったり……と，じつに忙しい処理になってくる。そのため，PICは充放電の監視やサウンドの管理だけに時間を取れなくなってくる。

一方，サウンド出力や明るさの測定はタイミングが命だ。サウンド出力については圧電素子にかける電圧の切り替え間隔をキチンと揃えないと音程が不安定になるし，明るさ測定については充放電の時間的条件を一律にしないと測定値が狂ってしまう。

そこで，今回必要とされる処理を，タイミング的に優先度の高い処理と，優先度の低い処理（ちょっとくらい遅れても問題のない処理）に分け，それぞれの優先度に応じて違う方法でプログラミングしよう。処理の内訳をまとめると**第1表**のようになる。

タイマ割り込みを使う

タイミングを厳密にしたい処理を行うのにはタイマ割り込みが便利だ。一般に「割り込み」とは，ある条件を満たしたときに現在実行中のプログラムを中断して，必要な処理を行う機能のことをいう。PICは何種類かの割り込み機能を持っているが，タイマ割り込みはそのうちのひとつだ。これは一定の経過時間を設定し，その時間が経過すると割り込みがかかる機能だ。割り込みがかかると，PICは現在実行中のプログラムを中断し，割り込みベクタ（PICではプログラムメモリの4番地）に自動的にジャンプする。ここで必要な処理を行い，最後にretfie命令で割り込みルーチンを終了すると，その後は先ほど中断したところから処理を再開する。

タイマ割り込みを使用するにはTMR0というタイマレジスタを操作する。TMR0は命令実行サイクルに同期して加算されていく特殊レジスタだ。今回のプログラムでは，TMR0は1命令実行のたびに1加算（インクリメント）されていくように設定してある。そしてその値がオーバフロー（1バイトのレジスタなので，255に1を足したときにケタ溢れが起き，0になる）するとタイマ割り込みフラグT0IFが1にセットされる。このとき，タイマ割り込みを許可しておけば，タイマ割り込みが発生する。PICは起動時にはすべての割り込みが禁止されているので，INTCONレジスタ（割り込みを管理する特殊レジスタ）のT0IE（タイマ割り込み許可）フラグをセットし，予め割り込みを許可しておく。

割り込み処理のたびにT0IFを0にリセットし，TMR0の値を初期化して，つぎの割り込みの準備をする。割り込みの間隔を短くするには，初期値を大きくすればよい。そうすればオーバフローまでの時間が短くなるからだ。今回は一定間隔で割り込みルーチンが自動的に呼びだされる，と思ってくれればいい。

割り込みルーチン vs メインルーチン

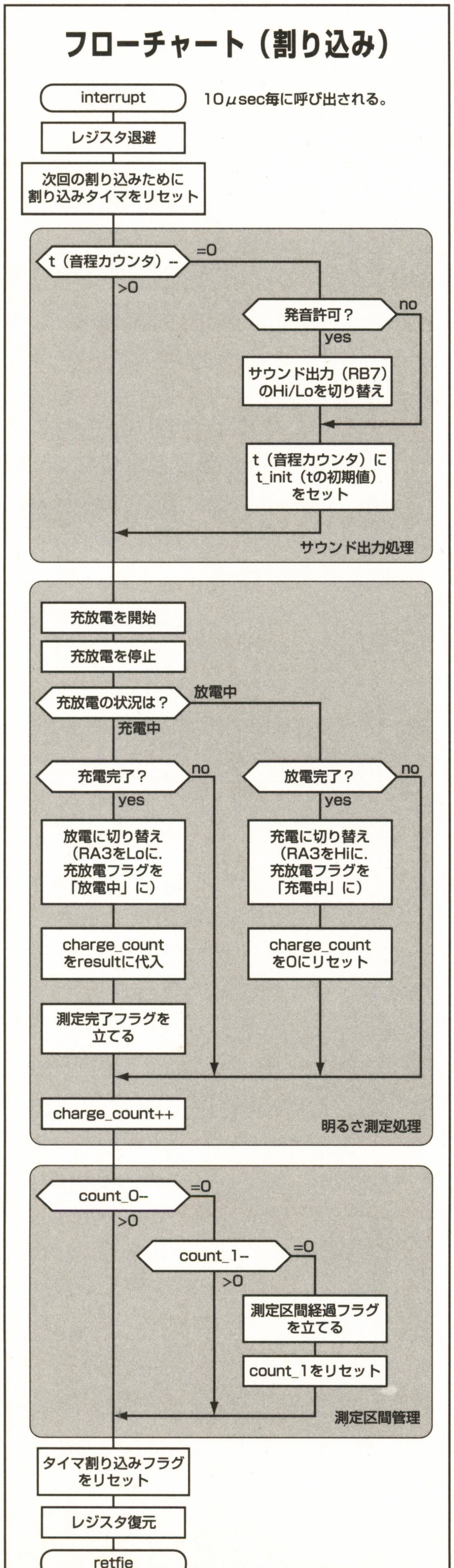
割り込みを使うときに注意が必要なのは，いつどこで割り込みルーチンが呼びだされるか分からないということだ。割り込みルーチンはメインルーチンで使っているレジスタの内容を破壊しないように気をつけなければならない。そのため，割り込みルーチンとメインルーチンで書き込むレジスタを明確に分けるようにする。一方が書き込むレジスタなら，他方では読んでもかまわないが，書き込むことは避ける。両方のルーチンで使うレジスタ（WレジスタやSTATUSレジスタなど）の場合は，割り込みルーチンの最初で内容を保存し，割り込み終了時に以前の値に戻してやる必要がある。

割り込みルーチンとメインルーチンの間で情報をやり取りするには，フラグを使用するとよい。

割り込みのプログラミング

第3図，**第4図**が今回コーディングする

プログラムのフローチャートだ。interruptから始まるのが割り込みルーチンで，これは正確に10μsec毎に呼びだされる。割り込みルーチンの中では，充放電のチェックとサウンド出力，それから測定区間の管理をしている。

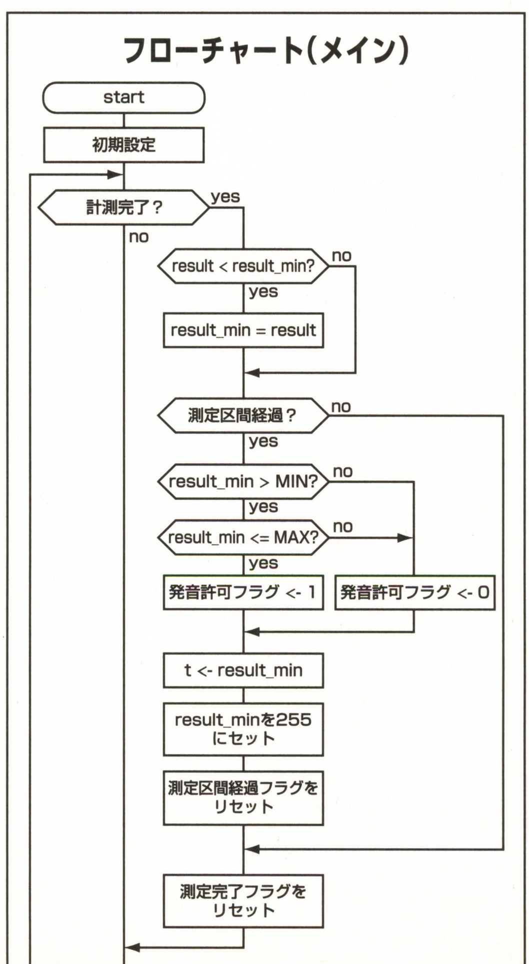


《第3図》改良版テミンの割り込みルーチンのフローチャートはこのようになる

充放電は割り込みルーチンの中でのみ行い，メインルーチン実行中は停止させる。割り込みがかかるたびにチョットずつ充放電をして，その都度完了したかチェックしていく。チェック回数はcharge_countに記録され，これは割り込みごとにインクリメントされる。charge_countは充電開始時に0にリセットされ，充電完了時にresultに保存される。1回の測定が完了したら，AD_DONEフラグを1に立ててメインルーチンに知らせる。

サウンド出力については，tというレジスタを使用する。tはあらかじめt_initの値で初期化しておき，割り込みごとに1減算（デクリメント）する。tが0になったら圧電素子にかける電圧のHi/Loを切り替える。同時にtを再びt_initで初期化する。

測定区間の管理には，2バイトのカウンタ（count_0とcount_1のセット）を使う。割り込みごとにデクリメントし，0になったらPERIODフラグを1に立て，1測定区間が経過したことをメインルーチンに知らせる。



《第4図》これがメインルーチンのフローチャートだ

メインルーチンのプログラミング

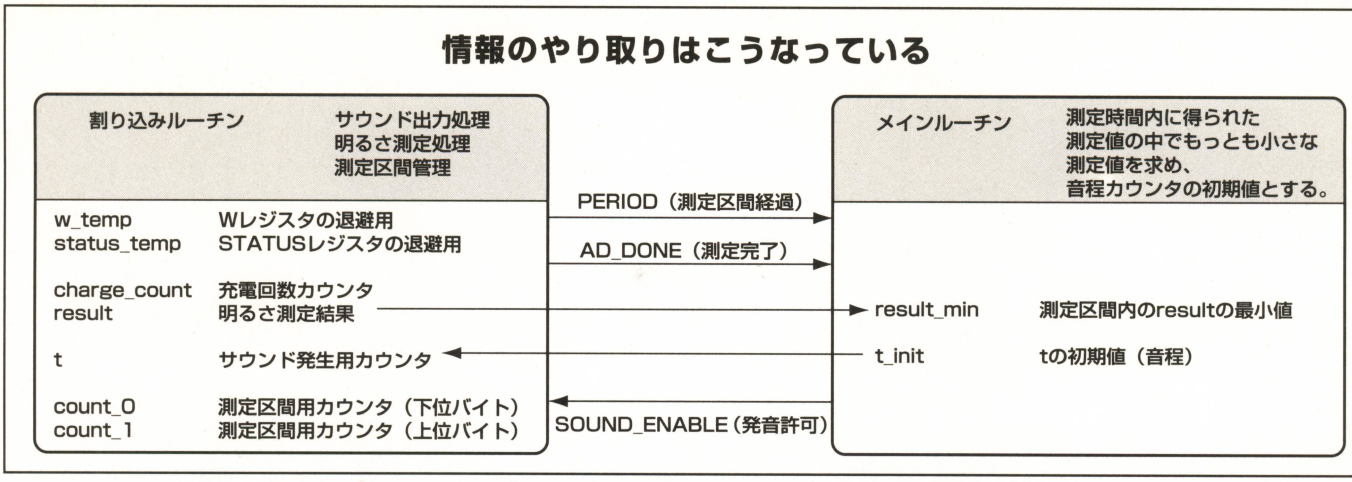
メインルーチンは，割り込みルーチンからの二つの通知をチェックしている。一つはAD_DONEフラグ，つまり測定完了のフラグだ。このフラグが立った場合，明るさの測定結果resultを読みだし，今までの測定値の最小値result_minと比較する。resultがresult_minより小さい場合はresult_minを更新する。

一方，もう一つのフラグ，PERIODが立っていた場合，測定区間の終了を意味するので，今までの測定値の最小値result_minが確定したことになる。そこで，この値がサウンド出力する範囲内にあるかどうか判断し，そうであれば発音許可フラグSOUND_ENABLEを1にセット，そうでなければこのフラグを0にセットしてミュートする。その後，result_minの値を，音程を示すt_initに書き込む。そして，つぎの測定区間のためにresult_minを255（＝1バイトの数値でもっとも大きい値）で初期化する。

処理完了時にAD_DONEフラグやPERIODフラグを0にリセットしていることに注意。これをしないと，何度も同じ処理をする羽目になってしまうからね。

このように，メインルーチンは割り込みルーチンが通知した各種の出来事をチェックし処理するという，イベント・ドリヴン型のプログラムになっている。**第5図**にそれぞれのルーチンが使用しているレジスタ名と，お互いの情報のやり取りについてまとめておいた。こういう「マンガ」を書いておくと，プログラミングやデバッグがしやすくなるぞ。

リスト1が今回のソースリストだ。例によってフロー・チャートと見比べて賞味していただきたい。前回と同じようにアセンブル後，HSモード，ウォッチドッグ不使用，パワーアップタイマ使用を選んでPICに書き込もう。書き込み完了



《第5図》割り込みルーチンとメインルーチンで使用するレジスタと，やり取りされる情報はこのようになっている

したら，PICをテミンのICソケットにさし込み，電源投入！　正常に動作していれば，極端に明るい所と暗い所では音が止まり，その中間の明るさでは音程が変化していくはずだ。それからパソコンやテレビのブラウン管にセンサを当てて音程が変わることも確認してほしい。

まとめ

今回のようなタイマ割り込みによるリアルタイム性の高い処理はマイコンならではのものだと思う。このようなテクニックを使うと，シリアル通信なども簡単にできるようになる。ちょびっと難しいけど，マスタすれば見返りは大きいぞ！

次回こそはケースに収めて仕上げをしたいと思う。それでは！

《リスト1》改良版テミンのアセンブラソース

```
;==================================================================
;
;                       Tемeн(テミン)
;                           rev 2.0
;                  coded by クワクボ リョウタ
;
;                  プロセッサ     : PIC16F84-20
;                  システムクロック : 20MHz
;                  ヒューズ       : HSモード
;                                  ウォッチドッグタイマ不使用
;                                  パワーアップタイマ使用
;
;==================================================================

        LIST    C=120, p=16F84, r=dec, x=off, n=0
        include <p16f84.inc>

;======== シンボルの定義 ===========================================
;-------- レジスタ名の定義 ------------
w_temp          equu    0x0c            ; Wレジスタの退避用
status_temp     equ     0x0d            ; STATUSレジスタの退避用
charge_count    equ     0x0e            ; 充電回数カウンタ
result          equ     0x0f            ; 明るさ測定結果
t               equ     0x10            ; サウンド発生用カウンタ
count0          equ     0x11            ; 測定区間用カウンタ（下位バイト）
count1          equ     0x12            ; 測定区間用カウンタ（上位バイト）
stat            equ     0x13            ; ステータスフラグ群
event           equ     0x14            ; イベントフラグ群
result_min      equ     0x15            ; 測定区間内のresultの最小値
t_init          equ     0x16            ; tの初期値（音程）

;-------- PORTAのビットマップ ----------
CHARGE_PIN      equ     3               ; 充放電ピン
READ_PIN        equ     4               ; 電圧チェックピン

;-------- PORTBのビットマップ ----------
BUZZER          equ     7               ; サウンド出力ピン

;-------- statのビットマップ -----------
CHARGING        equ     0               ; 0：放電中／1：充電中
SOUND_ENABLE    equ     1               ; 0：ミュート／1：発音許可

;-------- eventのビットマップ ----------
AD_DONE         equ     0               ; 1：測定終了
PERIOD          equ     1               ; 1：測定区間経過

;-------- 各タイマの初期値 -------------
TMR0_INIT       equ     215             ; TMR0の初期値
COUNT1_INIT     equ     10              ; count_1の初期値

;-------- 発音させる範囲 ---------------
MIN             equ     10              ; 明るい方のリミット
MAX             equ     245             ; 暗いほうのリミット

;========  リセットベクタ ==========================================
        org     0x0000                  ; 起動時は0番地から実行を始める
        goto    initialize

;========  割り込みベクタ ==========================================
        org     0x0004                  ; 割り込み時は4番地にジャンプする
;========  割り込みルーチン（10μsec毎に呼び出される） ==============
        movwf   w_temp                ; WレジスタとSTATUSレジスタの値を保存
        swapf   STATUS,W                ;
        bcf     STATUS,RP0              ;
        movwf   status_temp             ;

        movlw   TMR0_INIT               ; 割り込みタイマのリセット
        movwf   TMR0                    ; （次回の割り込みのために）

        ;-------- サウンド出力 ---------
        decfsz t,F                    ; サウンドの1周期が経過したら以下へ
        goto    int_30                  ; そうでなければint_30へ
        movlw   1<<BUZZER               ; サウンド出力ピンの電圧をトグル
        btfsc   stat,SOUND_ENABLE       ; （ミュートならスキップ）
        xorwf   PORTB,F                 ;

        movf    t_init,W                ; サウンド周期カウンタのリセット
        movwf   t                       ;

int_30:
        ;-------- 明るさの測定 ---------
        bsf     STATUS,RP0              ;
        bcf     TRISA,CHARGE_PIN        ; 充放電開始
        nop                             ;
        bsf     TRISA,CHARGE_PIN        ; 充放電停止
        bcf     STATUS,RP0              ;

        btfss   stat,CHARGING           ; 放電中ならint_40へ
        goto    int_40                  ; 充電中なら次へ

        ;-------- 充電中の場合 ---------
        btfss   PORTA,READ_PIN          ; 充電完了なら以下へ
        goto    int_50                  ; そうでなければint_50へ

        bcf     PORTA,CHARGE_PIN        ; 放電に切り替え（ピンの電圧）
        bcf     stat,CHARGING           ; 放電に切り替え（ステータス）

        movf    charge_count,W          ; 測定カウンタの値をresultに保存
        movwf   result                  ;

        bsf     event,AD_DONE           ; 測定完了を通知

        goto    int_50                  ; int_50へ

int_40:
        ;-------- 放電中の場合 ---------
        btfsc   PORTA,READ_PIN          ; 放電完了なら以下へ
        goto    int_50                  ; そうでなければint_50へ

        bsf     PORTA,CHARGE_PIN        ; 充電に切り替え（ピンの電圧）
        bsf     stat,CHARGING           ; 充電に切り替え（ステータス）

        clrf    charge_count            ; 測定カウンタをリセット

int_50:
        incf    charge_count,F          ; 測定カウンタをインクリメント
        btfsc   STATUS,Z                ; ただし最大値は0xff
        decf    charge_count,F          ;

        ;-------- 計測時間の管理 -------
        decfsz count0,F                 ; 一段目カウンタ
        goto    int_90                  ;

        decfsz count1,F                 ; 二段目カウンタ
        goto    int_90                  ;

        movlw   COUNT1_INIT             ; 二段目カウンタをリセット
        movwf   count1                  ;

        bsf     event,PERIOD            ; 測定区間経過を通知

int_90:
        ;------- 割り込み終了処理 -------
        bcf     INTCON,T0IF             ; タイマ割り込みフラグをリセット

        swapf   status_temp,W           ; WレジスタとSTATUSレジスタの値を復元
        movwf   STATUS                  ;
        swapf   w_temp,F                ;
        swapf   w_temp,W                ;

        retfie                          ; 割り込みルーチンからリターン

;========  初期設定  ===============================================
initialize:
        clrf    PORTA                   ; PORTAの出力を全部Loにする
        clrf    PORTB                   ; PORTBの出力を全部Loにする

        bsf     STATUS,RP0              ;

        movlw   b'00010000'             ; PORTAの入出力方向
        movwf   TRISA                   ; (1は入力、0は出力)

        movlw   b'00000000'             ; PORTBの入出力方向
        movwf   TRISB                   ; (1は入力、0は出力)

        movlw   b'01011000'             ; TMR0の設定
        movwf   OPTION_REG              ; （1命令サイクルごとにインクリメント）
        ;
        bcf     STATUS,RP0
        ;
        movlw   COUNT1_INIT             ; 測定区間カウンタの初期化
        movwf   count1                  ;

        clrf    stat                    ; statの初期化

        clrf    event                   ; eventの初期化

        bsf     INTCON,T0IE             ; タイマ割り込みを許可
        bsf     INTCON,GIE              ;
        ;
;========  メインルーチン  =========================================
main:
        btfss   event,AD_DONE           ; 測定完了なら以下へ
        goto    main                    ; そうでなければ最初に戻る

        movf    result,W                ; result <= result_minなら
        subwf   result_min,W            ; result_minを更新
        btfss   STATUS,C                ;
        goto    main_10                 ; そうでなければmain_10へ
        movf    result,W                ;
        movwf   result_min              ;

main_10:
        btfss   event,PERIOD            ; 測定区間経過なら以下へ
        goto    main_30                 ; そうでなければmain_30へ

        movf    result_min,W            ; result_min > MIN なら以下へ
        sublw   MIN                     ;
        btfsc   STATUS,C                ;
        goto    main_20                 ; そうでなければmain_20へ

        movf    result_min,W            ; result_min <= MAX なら以下へ
        sublw   MAX                     ;
        btfss   STATUS,C                ;
        goto    main_20                 ; そうでなければmain_20へ

        bsf     stat,SOUND_ENABLE       ; 発音許可
        goto    main_21                 ;
main_20:
        bcf     stat,SOUND_ENABLE       ; 発音停止（ミュート）
main_21:
        movf    result_min,W            ; 音程の更新
        movwf   t_init                  ;

        movlw   255                     ; result_minをリセット
        movwf   result_min              ;

        bcf     event,PERIOD            ; 測定区間経過フラグをリセット

main_30:
        bcf     event,AD_DONE           ; 測定完了フラグをリセット

        goto    main                    ; 最初に戻る

;==================================================================
        end
```

プログラミング
PROGRAMMING

プログラムの疑問・質問にお答えします

GET&RUN（ゲット ラン）

TEXT & PROGRAMMING●山崎 寛

アルゴリズム全般

VB

Q おはつです。最近プログラムをはじめましたので，イマイチ概要をつかめていません。

『ロックマン』風のゲームが作りたいのですが，参考になるゲームのソースがほしいです。よろしくお願いいたします。

なお，開発言語は『Microsoft Visual（マイクロソフト ビジュアル） Basic（ベーシック）（以下VBと略）6.0』です。

【神奈川県横浜市・M.K？歳】

A ロックマン風のゲームということは，サイドビューのジャンプ・アクション・ゲームと考えればいいでしょうか？

それなら，本誌2000年12月号の特選投稿プログラムに，『キャロット牧場』というよくできたジャンプ・アクション・ゲームがあります。参考になるのではないでしょうか。

後は，VBのプログラムではありませんが，2000年2月号の連載記事『超入門！ゲーム・アルゴリズム』では，サイドビューのゲームについて扱っています。このアルゴリズムも参考になるでしょう。

その他の連載記事でも，『チャレンジ!! ワンダーウィッチ』の2001年1～3月号，『catlikeのもっと身近にプログラミング！』などの記事も，アクション・ゲームの作成には参考になると思います。

アクションゲームの動きを考える

さて，上記と重複する部分もありますが，VBでのアクション・ゲームについてちょっと解説することにしましょう。

まず，キャラクターの動きについて考えてみましょう。ジャンプ・アクション・ゲームの動きでよくあるパターンは，左右は普通に移動して，ボタンでジャンプ，床のない場所では自由落下という形ですね（**第1図**）。

このような動きをもっと細かく見ると，

- **左右移動は，キーボードの入力からそのまま左右へと動かす**
- **床に足がついた状態でジャンプのキーが押されたら，ジャンプ開始**
- **床に足がついていない状態では，だんだん落下する**

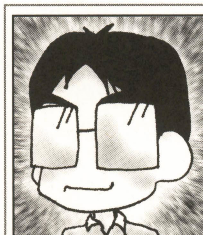
といった構造となります（**第2図**）。

ジャンプと落下については，キャラクターの縦方向の速度を表す変数を，あらかじめ用意しておきます。そして，ジャンプ開始時には，上方向（マイナス方向）へ大きな値を代入します。地面に足がついていないときは，つねにこの変数を下向き（プラス方向）に少しずつ値を足していくようにすることで，重力によって落下するという，物理法則にわりとそった動きを再現できます（**第3図**）。

これをVBで簡単に表現したのが，**リスト1**です。カーソル・キーの左右とスペース・キーで，画面内の四角を動かします。床などの地形は用意していないため，着地しているかどうかの判断は単純にY座標が160未満かどうかで判定しています。

VBで「現在キーが押されている状態」を知る方法は，いくつかあります。このプログラムでは，フォームのKeyUpやKeyDownイベントに応じて，キーの状態を示す変数keysを

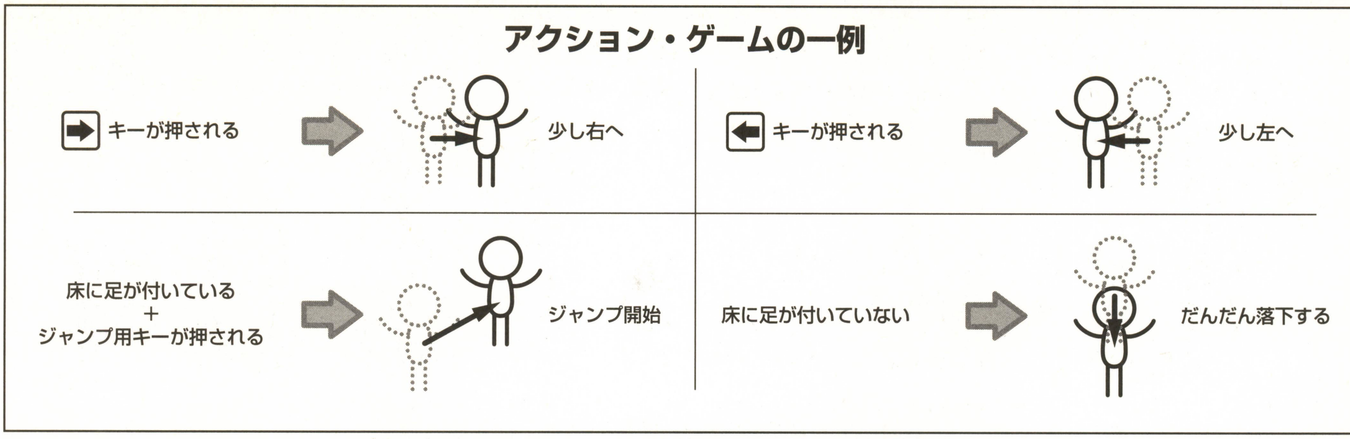
PROFILE



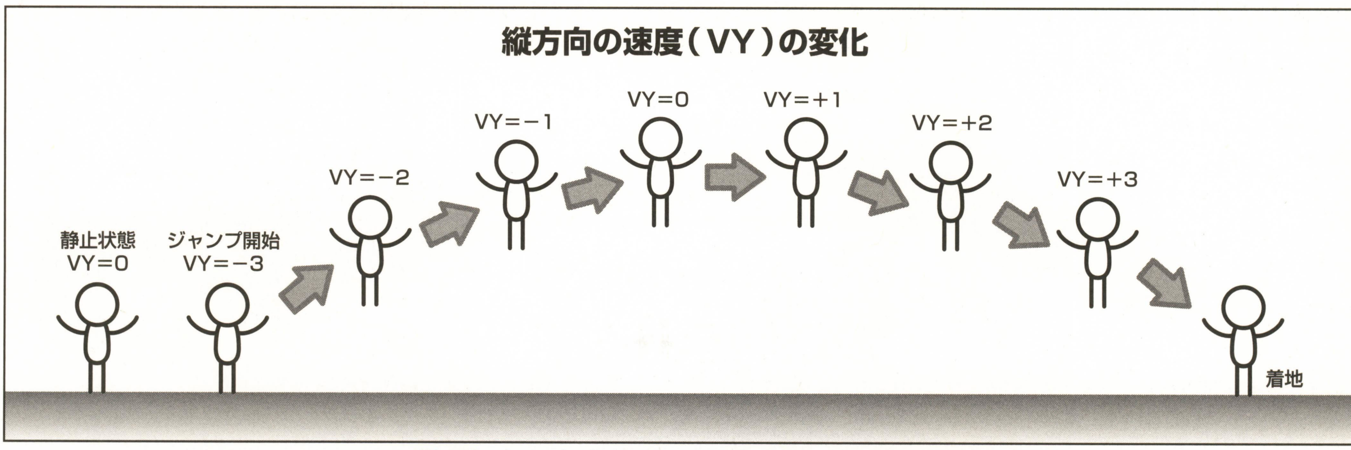
『M5』や『MSX』でプログラミングを覚えた，BASIC歴15年の男。コナミのゲーム・ミュージックを昔から好きということもあり，『ビーマニポケットKONAMIX』という携帯ゲームを購入。驚いたのが，音質のよさと収録曲のアレンジ。『グラディウス』『ツインビー』『悪魔城ドラキュラ』ってG.M.O.時代のCDによく似てない？ 古くからのゲーム・ミュージック好きの人がかかわったのかな？



《第1図》左右移動，ジャンプ，落下という動きは，アクション・ゲームでよくあるよね



《第2図》アクション・ゲームの動きを細かく見ると，このような対応がある



《第3図》ジャンプと落下の動きは，速度の変数をうまく使うのだ

変化させています。

仮想画面で地形の状態をチェック！

アクション・ゲームでは，地形を“マップデータ（仮想画面）”として扱うのが，衝突判定の容易さなどからも有利です。

マップデータについてのくわしい内容は，本誌2001年3月号の本コーナーや『catlikeのもっと身近にプログラミング！』を読み返してもらうとわかりやすいと思います。要は，配列に地形などの情報をブロック単位で入れておき，キャラクターの位置から対応する配列の場所を参照することで，床や壁の有無がわかるというものです。なお，キャラクターの移動が仮想画面の1ブロックより細かい場合で，キャラクターがブロックの中間に位置するときは複数のブロックを調べることになります。

リスト2は，マップデータを使い，ブロックが置かれた画面の中をジャンプと左右で動きまわるというサンプル・プログラムです（**写真1**）。実際にゲームにするにはもう少し改良したほうがいいのですが，参考になるはずです。

VB

計算方法全般

Q はじめまして，はじめて質問します。

今，『F-BASIC97』で作成したグラフ作成プログラムを，VB5.0に移植しようと考えているのですが，VBははじめてなので機能を十分把握していません。どうぞご教授願います。

①グラフ表示について

グラフを表示（または印刷）した場合，縦軸に90度回転させた文字を使用したいの

《リスト1》左右移動とジャンプのサンプル（VB用，フォームにタイマーコントロール一つを配置する）

```
Dim x As Integer, y As Integer 'キャラクターの位置
Dim vy As Integer '縦方向の速度
Dim mj As Integer 'ジャンプのフラグ
Dim keys(40) As Integer 'キー入力用変数

Private Sub Form_KeyUp(KeyCode As Integer, Shift As
Integer)
  'キーが離された処理
  If KeyCode <= 40 Then keys(KeyCode) = False
End Sub

Private Sub Form_KeyDown(KeyCode As Integer, Shift As
Integer)
  'キーが押された処理
  If KeyCode <= 40 Then keys(KeyCode) = True
End Sub

Private Sub Form_Load()
  '初期設定
  'フォームの単位をピクセルにする
  Me.ScaleMode = 3
  'タイマーの時間を設定
  Timer1.Interval = 50
  Timer1.Enabled = True
  '変数XYの初期値を設定
  x = 100: y = 160
End Sub
Private Sub Timer1_Timer()
  '前に描いた四角を消す
  Me.Line (x, y)-Step(31, 31), Me.BackColor, BF

  'キー入力に応じて左右に移動
  If keys(vbKeyLeft) Then x = x - 5
  If keys(vbKeyRight) Then x = x + 5

  '着地状態でスペースが押されたらジャンプ開始
  If mj = False And keys(vbKeySpace) Then vy = -16: mj =
True

  '縦方向の位置をずらす
  y = y + vy

  '着地状態じゃなければVYを+1する
  If y < 160 Then
    vy = vy + 1
  Else
    vy = 0: mj = False
  End If

  '新しく四角を描く
  Me.Line (x, y)-Step(31, 31), &HFF, BF

End Sub
```

《リスト2》ブロックが配置されている中を動きまわるサンプル（VB4.0〜VB6.0用，フォームにタイマーコントロールを一つ配置する）

```
'APIの宣言
Private Declare Function BitBlt Lib "gdi32" (ByVal hDestDC
As Long, ByVal x As Long, ByVal y As Long, ByVal nWidth As
Long, ByVal nHeight As Long, ByVal hSrcDC As Long, ByVal
xSrc As Long, ByVal ySrc As Long, ByVal dwRop As Long) As
Long

'変数の宣言
Dim map(19, 14) As Integer '仮想画面用配列
Dim mx As Integer, my As Integer 'キャラクターのXY座標
Dim vx As Integer, vy As Integer 'XY彷徨の移動速度
Dim mj As Integer 'ジャンプ用フラグ
Dim keys(40) As Integer 'キー入力用
Const max_vx = 6, max_vy = 16 '移動と落下の最高速度

Private Sub Form_KeyUp(KeyCode As Integer, Shift As Integer)
  If KeyCode <= 40 Then keys(KeyCode) = False
End Sub

Private Sub Form_KeyDown(KeyCode As Integer, Shift As
Integer)
  If KeyCode <= 40 Then keys(KeyCode) = True
End Sub

Private Sub Form_Load()
'初期設定
  Me.ScaleMode = 3
  Me.Width = 9600: Me.Height = 7600
  Me.Move 0, 0
  Picture1.ScaleMode = 3
  Picture1.Width = 640: Picture1.Height = 480
  Picture1.Visible = False
  Picture1.BorderStyle = 0
  Picture1.AutoRedraw = True

  'タイマーの時間を設定
  Timer1.Interval = 50
  Timer1.Enabled = True

  '仮想画面にブロックを書き込む
   For i = 0 To 19: For j = 0 To 14: map(i, j) = 1: Next j:
Next i
  For i = 1 To 18: For j = 1 To 13: map(i, j) = 0: Next j:
Next i
  For i = 0 To 50: x = Int(Rnd(1) * 20): y = Int(Rnd(1) *
15): map(x, y) = 1: Next i

  '仮想画面にしたがってピクチャボックスの内容を描く
  For j = 0 To 14
    For i = 0 To 19
      If map(i, j) = 0 Then
             Picture1.Line (i * 32, j * 32)-Step(32, 32),
&HFFFFFF, B
         Picture1.Line (i * 32 + 1, j * 32 + 1)-Step(30, 30),
&H0, BF
      Else
             Picture1.Line (i * 32, j * 32)-Step(32, 32),
&HFFFFFF, BF
      End If
    Next i
  Next j

End Sub

Private Sub Timer1_Timer()
'メインルーチン

    '左右移動
    If keys(vbkeyleft) Then
      '左が押されたので，移動速度VXを左方向に加速
      vx = vx - 1: If vx < -max_vx Then vx = -max_vx
    ElseIf keys(vbkeyright) Then
      '右が押されたので，移動速度VXを右方向に加速
      vx = vx + 1: If vx > max_vx Then vx = max_vx
    Else
      '何も押されてなければVXを0になるように減速
      vx = vx - Sgn(vx)
    End If

    '移動速度VXにしたがってX座標を変更
    mx = mx + vx
    '仮想画面をチェックして壁があれば位置を補正する
     If map(mx ¥ 32, my ¥ 32) > 0 Or map(mx ¥ 32, (my + 31) ¥
32) > 0 Then
      '左がブロックに触れたのでブロックの右に
      '位置をずらし，速度を0に戻す
      mx = (mx ¥ 32) * 32 + 32: vx = 0
    End If
     If map((mx + 31) ¥ 32, my ¥ 32) > 0 Or map((mx + 31) ¥
32, (my + 31) ¥ 32) > 0 Then
      '左の逆の処理
      mx = (mx ¥ 32) * 32: vx = 0
    End If

    'ジャンプフラグとスペースキーの状態でジャンプ開始
    If keys(vbkeyspace) And mj = False Then vy = -16

    'Y座標をVYの値にしたがって変更
    my = my + vy
    'VYの値を増やして下方向に加速
    vy = vy + 1: If vy > max_vy Then vy = max_vy

    '仮想画面をチェックする
     If map(mx ¥ 32, my ¥ 32) > 0 Or map((mx + 31) ¥ 32, my ¥
32) > 0 Then
      '頭上のブロックにぶつかったのでVYを0にして落下させる
      my = (my ¥ 32) * 32 + 32: vy = 0
    End If
     If map(mx ¥ 32, (my + 31) ¥ 32) > 0 Or map((mx + 31) ¥
32, (my + 31) ¥ 32) > 0 Then
      '足がブロックに触れたので落下を止めジャンプフラグをリセ
ット
      my = (my ¥ 32) * 32: vy = 0: mj = False
    Else
      '床に触れていない間はジャンプフラグを立てる
      mj = True
    End If

    'ピクチャボックスの内容をフォームに転送
     ret = BitBlt(Me.hDC, 0, 0, 640, 480, Picture1.hDC, 0, 0,
vbsrccopy)
    'キャラタター（赤い四角）を描く
    Line (mx, my)-Step(31, 31), &HFF, BF

End Sub
```

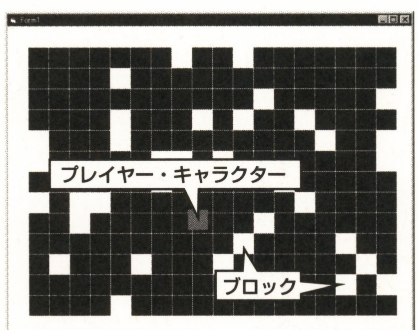
ですが，VBにはsymbolのような命令がないようなので困っています。Windows APIのCreateFontを使用すれば，画面に表示することは可能なようですが，印刷の場合はどうしたらよいのかわかりません。

基本的に印刷は，Printer命令を使用しています。何かよい方法はないでしょうか？

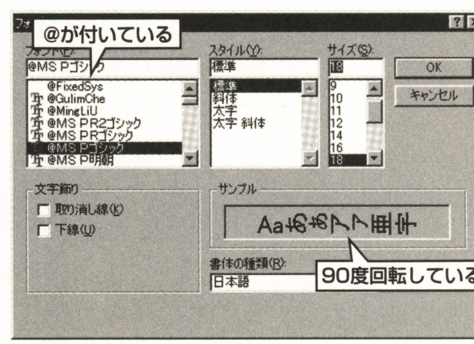
②計算誤差をなくしたい

計算値の誤差を少なくしたいので，『BCD演算』があれば助かるのですが…。VBにこれがない場合，計算誤差を回避できる方法，または誤差を少なくするために気をつけることがありますか？

【長崎県松浦市・風竜汰26歳】



《写真1》リスト2の実行画面。ゲームではないが，壁や床にぶつかる動きを体験できる



《写真2》「@」の付いた名前のフォントは縦書き用のため，もともと回転しているのだ

A それでは，一つずつ答えていきましょう。

①グラフ表示について

文字を90度回転させたいということですが，Windowsには「縦書き用フォント」というのがあるのをご存知でしょうか？

「@MS ゴシック」というように，頭に@（アットマーク）記号が付いてるのが特徴です。**写真2**のように，あらかじめ文字が反時計まわりに90度傾いています。

このようなフォントを使えば，簡単に90度回転した文字を画面やプリンタに出力できます。ただし，当然ながら反時計回りの90度に限定されますし，もともとが縦書き用のため，英数字はそのままになってしまうという点に注意が必要です。

縦書きフォントでは不満な場合，普通の文字を回転させて描くことになります。この場合は，質問に書かれているようにCreateFontなどのWindows APIを利用する方法があります。

CreateFontを使って回転した文字を表示する場合，CreateFont, SelectObject, DeleteObject，TextOutの四つのWindows APIを使います。

手順としては，

①CreateFontで論理フォント作成
②SelectObjectでフォントを割りあてる
③TextOutで文字を表示
④DeleteObjectでオブジェクトを解放

といった流れになります。Windows APIの宣言などは，サンプル・プログラムを参考にしてください。

つぎに，各Windows APIの使いかたを紹介しましょう。**第1表**も参照してください。

CreateFontは引数が多く，一見複雑そうに見えますが，実際はそれほどいじる必要のある場所はないでしょう。CreateFontの引数は，最初から順番に**第1表**のようになっています。文字の角度はnEscapementとnOrientationで，0.1度単位です。そのため，90度ならば900を両方に指定します。後は，フォントの名前と，必要ならば太さや斜体，下線などを指定するといいでしょう。

SelectObjectとDeleteObjectの引数は，**第1表**のようになっています。DeleteObjectの引数は一つだけで，使い終わった論理フォントを渡してオブジェクトを解放します。

TextOutの引数で注意するのはnCountの数で，VBで日本語文字列を渡す場合，StrConvを使ってUnicodeから変換し，LenBで値を得ます。

以上の処理で，回転したフォントを表示できます。**リスト3**がそのサンプルで，ボタンを押すとフォーム上に90度回転した文字列が書かれます(**写真3**)。

《第1表》各Windows APIの引数

引数	意味
●CreateFont	
nHeight	フォントの高さ(0で省略可)
nWidth	フォントの平均幅(0で省略可)
nEscapement	テキスト行の角度(0.1度単位で指定)
nOrientation	ベースラインとx軸の角度(nEscapementと同じ値を入れる)
fnWeight	フォントの太さ(0で省略可)
fdwItalic	斜体属性(0以外でtrue)
fdwUnderline	下線属性(0以外でtrue)
fdwStrikeOut	打ち消し線属性(0以外でtrue)
fdwCharSet	文字セット識別子(日本語Shift-JISの場合128)
fdwOutputPrecision	出力精度(0で標準)
fdwClipPrecision	クリッピング精度(0で標準)
fdwQuality	出力品質(0で標準)
fdwPitchAndFamily	ピッチとファミリ(どちらも0で標準)
lpszFace	フォントの名前文字列
●SelectObject	
hdc	出力先となるフォームやピクチャボックスのhDCプロパティを指定
hObject	CreateFontで作成した論理フォントを指定
●DeleteObject	
hObject	論理フォントを指定
●TextOut	
hdc	出力先のhDC
x	表示する文字のX座標
y	表示する文字のY座標
lpString	表示する文字列
nCount	表示する文字数(バイト単位)

印刷はどうする？

つぎに印刷ですが，本来のWindowsの仕組みでは，フォームもプリンタも同等に扱えます。ですから，単純にプログラム上で，フォームのhDCを指定している部分をPrinterオブジェクトのhDCに変えればそのままで印刷できるはずです。

なんで「本来」とか「はずです」と書いたかというと，どうやらVBのバグのように思えるのですが，VB4.0では正常に印刷できるものの，VB6.0ではできません(VB5.0ではランタイムのバージョンによって異なるようです)。これは，VB5.0で印刷機能のバグを修正する際の副作用でなってしまったようです。

ということで，結論としてはVB5.0／VB6.0では，Printerオブジェクトを用いた印刷で文字を回転させることはできません。

どうしても印刷したい場合，「CreateDCなどのWindows APIを用いてデバイスコンテキストを作成し，印刷処理を自前でやる」という，けっこうめんどうな形になります。**リスト4**がそのサンプルですが，これは本当に印刷処理の最低限の部分で，実用的なアプリケーションの場合はもう少し複雑になるでしょう。

一応Windows APIでの印刷処理について簡単に説明しておくと，Printerオブジェクトから現在デフォルトに設定されているプリンタのデバイス名を取得し，Windows APIのCreateDCによってプリンタのデバイスコンテキストを作成してhDCを得ます。作成したデバイスコンテキストに対してプリントを開始するよう設定したら，後は描画処理をして改ページなり印刷終了を指示すれば，実際にプリンタから印刷結果が出力されます。

先ほどの回転させた文字を出力する部分では，Windows APIのTextOutによって指定したデバイスコンテキストに対して文字を出力していましたから，ここでは作成したプリンタのデバイスコンテキストを指定するようにして印刷しているわけです。

ただし，このようにしてWindows APIで印刷処理を行う場合，Printerオブジェクトを使用することはできないため，すべての描画処理はWindows APIを用いて直接デバイスコンテキストに出力する必要があります。まあ，VBでの描画命令のほとんどはたいていWindows APIでも置き換えられると思いますので，ちょっとめんどうかもしれませんがムリではないでしょう。

後は，印刷が終了したら，作成したhDCを忘れずに解放するようにしてください。Windows APIを使う場合，「開い

《リスト3》CreateFontで回転した文字をフォームに表示するサンプル(VB4.0～VB6.0用，フォームにコマンドボタンを一つ配置する)

```
Private Declare Function CreateFont Lib "gdi32" Alias
"CreateFontA" (ByVal H As Long, ByVal W As Long, ByVal E As
Long, ByVal O As Long, ByVal W As Long, ByVal I As Long,
ByVal u As Long, ByVal S As Long, ByVal C As Long, ByVal OP
As Long, ByVal CP As Long, ByVal Q As Long, ByVal PAF As
Long, ByVal F As String) As Long
Private Declare Function SelectObject Lib "gdi32" (ByVal hdc
As Long, ByVal hObject As Long) As Long
Private Declare Function DeleteObject Lib "gdi32" (ByVal
hObject As Long) As Long
Private Declare Function TextOut Lib "gdi32" Alias "TextOutA"
(ByVal hdc As Long, ByVal x As Long, ByVal y As Long, ByVal
lpString As String, ByVal nCount As Long) As Long

Private Sub Command1_Click()

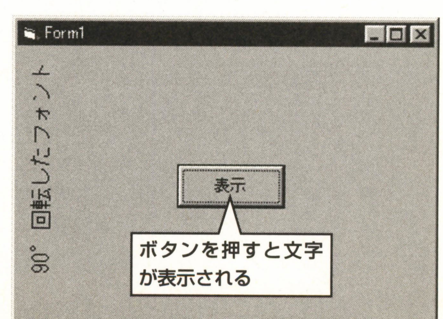
Dim hFont As Long  '作成した論理フォント用
Dim hOldFont As Long  '古い論理フォント保存用
Dim a$
Dim ret As Long

'CreateFontで90°回転したフォントを作成
hFont = CreateFont(0, 0, 900, 900, 0, 0, 0, 0, 128, 0, 0, 0,
0 Or 0, "ＭＳ ゴシック")

'フォームに作成したフォントを割り当てる
hOldFont = SelectObject(Me.hdc, hFont)
'フォームに文字を描く
a$ = "90°回転したフォント"
ret = TextOut(Me.hdc, 10, 170, a$, LenB(StrConv(a$,
vbFromUnicode)))

'フォームのフォントを元に戻す
ret = SelectObject(Me.hdc, hOldFont)
'オブジェクトを解放
ret = DeleteObject(hFont)

End Sub
```



《写真3》CreateFontを用いて，文字を回転表示しているサンプル実行画面

《リスト4》Windows APIで印刷処理をするサンプル（VB4.0～VB6.0用，フォームにコマンドボタンを一つ配置する）

```
'回転フォント用のAPI宣言
Private Declare Function CreateFont Lib "gdi32" Alias "CreateFontA" (ByVal H As Long, ByVal W As Long, ByVal E As Long, ByVal O As Long, ByVal W As Long, ByVal I As Long, ByVal u As Long, ByVal S As Long, ByVal C As Long, ByVal OP As Long, ByVal CP As Long, ByVal Q As Long, ByVal PAF As Long, ByVal F As String) As Long
Private Declare Function SelectObject Lib "gdi32" (ByVal hdc As Long, ByVal hObject As Long) As Long
Private Declare Function DeleteObject Lib "gdi32" (ByVal hObject As Long) As Long
Private Declare Function TextOut Lib "gdi32" Alias "TextOutA" (ByVal hdc As Long, ByVal x As Long, ByVal y As Long, ByVal lpString As String, ByVal nCount As Long) As Long
'印刷関係のAPI宣言
Private Declare Function CreateDC Lib "gdi32" Alias "CreateDCA" (ByVal lpDriverName As String, ByVal lpDeviceName As String, ByVal lpOutput As String, lpInitData As Any) As Long
Private Declare Function StartDoc Lib "gdi32" Alias "StartDocA" (ByVal hdc As Long, lpdi As DOCINFO) As Long
Private Declare Function StartPage Lib "gdi32" (ByVal hdc As Long) As Long
Private Declare Function EndPage Lib "gdi32" (ByVal hdc As Long) As Long
Private Declare Function EndDoc Lib "gdi32" (ByVal hdc As Long) As Long
Private Declare Function DeleteDC Lib "gdi32" (ByVal hdc As Long) As Long
'印刷用型宣言
Private Type DOCINFO
  cbSize As Long
  lpszDocName As String
  lpszOutput As String
End Type

Private Sub PrintOut()
'APIによるプリンタのhDC取得＆印刷
Dim devName$ 'プリンタのデバイス名
Dim lphPrinter As Long
Dim prhdc As Long
Dim ret As Long
Dim dinfo As DOCINFO

  'Printerオブジェクトからデフォルトのプリンタ名を
  '取得し，そのデバイスコンテキストを作成
  devName$ = Printer.DeviceName
  prhdc = CreateDC("WINSPOOL", devName$, "", vbNullString)

  'DOCINFO構造体の内容を設定
  dinfo.lpszDocName = "VB印刷テスト" 'ドキュメント名
  dinfo.lpszOutput = vbNullString
  dinfo.cbSize = LenB(dinfo) 'dinfoのサイズを入れる

  '印刷開始
  ret = StartDoc(prhdc, dinfo)
  ret = StartPage(prhdc)

  'hDCへの書きこみ処理を呼ぶ
  Call prt_sub(prhdc)

  '印刷終了
  ret = EndPage(prhdc)
  ret = EndDoc(prhdc)

   '作成したデバイスコンテキストを解放
  ret = DeleteDC(prhdc)

End Sub
Private Sub prt_sub(ByVal prhdc As Long)
'hDCへ縦の文字を書きこむ
Dim hFont As Long  '作成した論理フォント用
Dim hOldFont As Long  '古い論理フォント保存用
Dim a$
Dim ret As Long

  'CreateFontで90° 回転したフォントを作成
  hFont = CreateFont(0, 0, 900, 900, 0, 0, 0, 0, 128, 0, 0, 0, 0 Or 0, "ＭＳ ゴシック")

  '指定されたhdcに作成したフォントを割り当てる
  hOldFont = SelectObject(prhdc, hFont)
  '文字を描く
  a$ = "90° 回転したフォント"
    ret = TextOut(prhdc, 1000, 1070, a$, LenB(StrConv(a$, vbFromUnicode)))

  '変更したフォントを元に戻す
  ret = SelectObject(prhdc, hOldFont)
  'オブジェクトを解放
  ret = DeleteObject(hFont)

End Sub

Private Sub Command1_Click()
  Call PrintOut
End Sub
```

たものは閉じる」というのは鉄則です。

②計算誤差をなくしたい

計算の誤差についての質問ですが，ちょっと高度な内容なので，質問の意味がわからない人向けに説明をしておきましょう。

コンピュータというのは，内部では“2進数”で計算をしていることは，だいたいの人が知っていると思います。この2進数は，私たちが普段使っている10進数とでは“循環小数”になる値が変わってきます。循環小数というのは，たとえば10進数で「10÷3＝3.333333…」と，小数点以下に同じ数字が続きますね。このように，小数点以下にある数字の列が繰り返し続くときのことを指します。

10進数では「0.1」は循環小数ではありませんが，これを2進数にしようとすると「0.0001100110011…」と循環小数になってしまいます。

循環小数は無限に数字が続くため，桁数に限りのあるコンピュータでは誤差がでてしまいます。そのため，10進数では普通に表せる数同士の計算でも，コンピュータに計算させた場合誤差がでることがあります。

たとえば，VB上のイミディエイト・ウィンドウで

```
?1.1-1.0-0.1
```

なんて実行すると，表示されるのは本当は0のはずが，

```
8.32667268468867E-17
```

という結果になります。この数字は「限りなく0に近い数」ですが，人間にとって簡単な計算がちがう結果となるのは不思議に思いますよね。

このような誤差は，2進数で小数を表すことが原因となっているので，2進数か小数を使わないようにすれば防ぐことができます。

その方式の一つが，“BCD（Binary（バイナリ） Corded（コーデッド） Decimal（デシマル）：2進化10進法）”です。2進数4桁で10進数1桁を表し，コンピュータ内部でも10進数で計算するため，2進数が原因となる誤差を防げます。

また，“固定小数点方式”の場合も，内部的には小数を使わず整数として扱えるため，同様に誤差がでません。

VBでBCDや固定小数点を使うには

つぎに，VBでの扱いかたを紹介しましょう。前述したBCDと固定小数点は，ともにVBで扱うことが可能です。

まず固定小数点ですが，VBではCurrencyと呼ばれる変数の型があります。この型を使った場合，内部では固定小数点で格納されます。お金などの誤差が許されない用途向けという意味で，“通貨型”という名前が付いています。

この通貨型の変数を使うことで，誤差を減らすことができます。注意点としては，固定小数点なので小数点以下4桁までしか扱えないこと，扱える値の範囲が狭いこと，計算に他の型の変数を混ぜないことなどがあります。

つぎにBCDですが，BCDはちょっと特殊な使いかたをします。VB内部には，BCDで数値を扱うDecimalというデータ型があるのですが，変数をDecimal型として宣言することはできません。

そのため，BCDを使いたい変数は，すべてValiant型で宣言しておきます。そして，数値を扱う場合には，全部CDec関数を利用しDecimal型に変換して扱うようにします。こちらも注意点としては，他の型を混ぜないということです。

コンピュータによる計算には誤差がつきものですが，誤差の発生する原因や必要な精度などをつねに考えながら対策することが大事です。

BCDや固定小数点を使えば，2進数の小数が原因の誤差は防げますが，有効桁数など他の要因による誤差が発生する可能性もあります。どのような計算をするのか，扱う値の範囲はどうかなど，場合に合わせた対処ができるようにしましょう。

疑問・質問ないですか？

このコーナーでは，プログラムに関するQ&Aを掲載しています。

プログラムに関する疑問（「こんなゲームを作りたいんだけど，ここがうまくいかない！」とか「このプログラムの入力方法がわからない～」）などがありましたら，なるべく具体的に，使用機種や言語を明記して，下記住所までハガキや封書もしくは電子メールでお送りください。住所，氏名（ペンネーム），年齢，職業，電話番号も忘れずに。

■郵便あて先

〒141-8715　東京都品川区東五反田1-11-15　電波新聞社マイコンBASICマガジン編集部　GET＆RUN係

■電子メールあて先

bm@jolls.co.jp

※メールのタイトル／Subjectに「GET＆RUN係」と明記してお送りください。

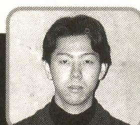
DTM Art Studio パソコンで音楽をたのしもう!

協 力
ヤマハ（http://www.yamaha.co.jp）

Text&SoundData
大川ゴウ／幸田英樹

このコーナーは，芸大生が体験したDTMの基本，応用，実践を紹介するコーナーです

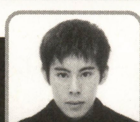
筆者紹介



大川ゴウ：

大阪芸大
芸術学部　音楽学科
音楽制作コース（作曲専攻）3回生

こんにちは～！新しい年度になり，学校に通っている方，お勤めの方，ともに気が引き締まる思いですね～。さあ，みなさん，目標をもって今年もがんばりましょう！僕も今年は室内楽の作曲を目標にがんばります。



幸田英樹：

大阪芸大
芸術学部　音楽学科
音楽工学コース（音楽工学専攻）3回生

こんにちは～。今，電子工学の勉強をしています。具体的に言えば，マイク，スピーカアンプなどを作るときの回路計算などです。この記事でも，ゆくゆくは簡単で低コストのマイク作成講座をする予定ですので，楽しみに待っていてください。

前回のおさらい

前回は，初心者のみなさんが溶け込みやすいように，めちゃらくソフトとソフトシンセサイザのインストールの説明と設定の仕方について解説しました。インストールに関しては，画面の指示どおりにしたがえば，まず問題はないでしょう。ソフトシンセサイザに関しても，ドライバ（簡単に言えば，めちゃらく作曲ソフトとソフトシンセサイザを結ぶものというところでしょうか。これがなければ，ソフトシンセサイザの音を鳴らすことはできません）を自動で換えてくれますし，まずこれも問題はないですね。

少し注意することは，再起動後のソフトを立ち上げたときにでる＜MIDI設定ダイアログ＞（出力先）という画面でYAMAHA SXG Driverを選択することぐらいですね！

さぁ～ここまで来たら今度は曲作りだっ！！

＜「DTM Art Studio」連動WEBサイトはこちら＞

http://www.beeplus.or.jp/bm/art/



トップページには，各協力企業へのリンク，マイコンBASICマガジンへのリンク，その他のページです。このページの中には，僕たち大阪芸術大学生のプロフィールや僕たちが「めちゃらく作曲名人」で作った曲のデータもDL（ダウンロード）できるようになっています。

その他，作品募集の方法やコンテストについてのくわしい情報が満載です。

この場所は，ライターと読者が一体となれることを目的としているので，質問や曲データの投稿も気軽に行えます。インターネットの環境が整っている人は，今すぐアクセスしてみてください！

■音楽のスタイル

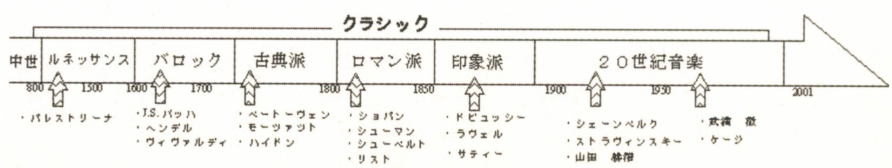
・クラシック音楽

みなさんはクラシック音楽と聴いてどんなイメージを持たれるでしょうか？聴いていて眠くなる音楽，退屈な音楽，難しくてわかりにくい音楽，カタブツで頭でっかちな人が聴く音楽などというイメージを持たれているんじゃないでしょうか？なかには聴いていて落ち着くα波の出る音楽と答える人もいるかもしれません。

筆者も数年前まではクラシック音楽は大嫌いですと答えていたと思います。しかし専門的に学ぶ今になって考えてみれば，ある時代のスタイルが肌に合わなかっただけだったことに気がつきます。

そもそもクラシック音楽とは，ヨーロッパを中心に発展した西洋音楽の総称なので，音楽の一スタイルではありません。そういう意味ではクラシック音楽の中にも，さまざまなジャンルやスタイルが存在するということです。みなさん一度だまされたと思って，下の表を参考に各時代の作品を鑑賞し，雰囲気の違いを感じてみてください。きっとあなたに合うスタイルが見つかるはずですよ！

ちなみに筆者は印象派の作品を好んで聴きます。



YAMAHA「めちゃらく作曲名人」～作曲編～

YAMAHA「めちゃらく作曲名人」の紹介

YAMAHA「めちゃらく作曲名人」は今までにはなかった新感覚作曲ソフトです。

難しい音楽知識などはまったく必要とせず，スロットマシンを使っての伴奏・メロディ・アレンジをしてしまうとってもユニークな発想から誕生したソフトです。

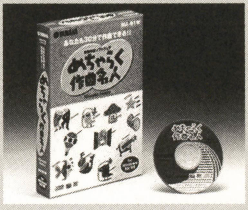
今まで曲を作れないでいた人はもちろんのこと，ゲーム好きな方々にも楽しんでお使いになれます。

製品仕様

●CD-ROM（「めちゃらく作曲名人」，ソフトシンセサイザ「S-YG20」，インターネットブラウザソフト用MIDIプラグインソフトウエア「MIDPLUG」，ヤマハCBXドライバー収録）
●取扱説明書

動作環境

●対応パソコン：NEC PC-98シリーズ，IBM-PC／AT並びにその完全な互換機
●対応OS：WindowsMe／98／95
●CPU：Pentium-100MHz以上（ソフトシンセサイザ使用時はMMX Pentium-166MHz以上）
●実装メモリ：32MB以上
●ハードディスク必要容量：60MB以上
●音源：同梱ソフトシンセサイザまたはXG対応MIDI音源機器
●ディスプレイ：800×600ドット，16ビットHigh Color以上
●必要機器：マウス，ハードディスク，2倍速以上のCD-ROMドライブ



9,800円

「めちゃらく作曲名人」を使っての簡単な作曲手順説明

・「めちゃらく作曲名人」の四つの画面の説明

音楽ソフトというと，五線譜とか音楽理論とか難しいことを考えがちです。ここでは，「めちゃらく作曲名人」がそういったソフトと対照的であることと，新しいソフトに対する初心者のかすかな拒否反応を軽減するため，曲作りの前の予備知識（全体の構造）について少し触れておきましょう。

まず，この「めちゃらく作曲名人」は，四つの画面で構成されています。これを聴いただけでも，このソフトがいかにシンプルであるかがわかります。さらに，このソフトの本当にすごいところは，30分で曲が作れてしまうというところにあります。

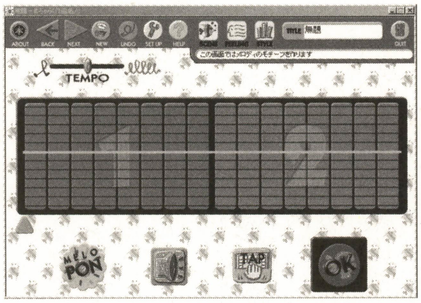
それでは画面説明です。

■バッキング画面



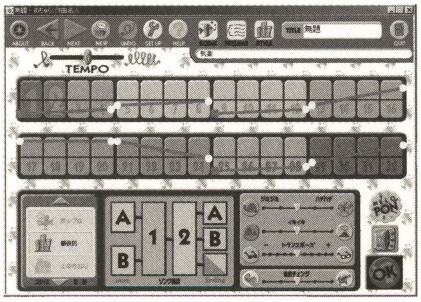
この画面がこの「めちゃらく作曲名人」を立ち上げたとき，最初に出てくる画面です。ここでは，名前のとおり伴奏のパターンを作ります。

■モチーフ入力画面



ここでは「バッキング画面」で作った伴奏に，メロディをつけるところです。

■ソング構成画面



ようやくここで，伴奏とメロディを用いて，曲を完成させます。

■メイクアップ画面



ここでは，出来上がった曲を，メールに添付して送信したり，自分のホームページで自作曲を鳴らすことができます。また，その曲を入れた，MD・カセット・フロッピーにかわいく包装することもできます。

・メニューバーと各ボタンの説明



1. ABOUTボタン



製品情報（バージョン）を表示します。

2. NEXT/BACKボタン



つぎの画面に行ったり，前の画面に戻ったりできます。

3. NEWボタン



途中で曲を作りなおしたくなったときに，新規作成ファイルを作成してくれます。

注意：作りかけている曲は保存されません。

4. UNDOボタン



失敗して実行した操作をやり直すことができます。クリックするたびに，過去の操作にさかのぼって取り消しますが，画面を移動するとやり直しができなくなります。

5. SETUPボタン



ドライバの設定を行います。メイクアップ画面ではメーラー（メールソフト）の設定を行います。

6. HELPボタン

オンラインヘルプを起動し，各画面の用語について，くわしく説明してくれます。



7. SCENE/FEELING/SYYLEアイコン



バッキング画面で選択された，シーン/フィーリング/スタイル/のイラストが表示されます。

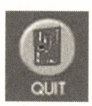
8. TITLE欄



曲のタイトルが表示されます（後で説明しますが，曲が完成された後でも変更可能です)。

9. QUITボタン

アプリケーションを終了します。



10. MELOPONボタン

あなたの代わりに，ランダムに伴奏やメロディ，曲の構成をつくってくれます。



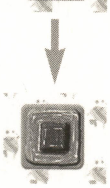
11. スピーカボタン

現在作成中の曲を聴きたいとき，これを押すと演奏してくれます。



12.ストップボタン

演奏を停止します。演奏がはじまると，スピーカボタンはこのストップボタンに変わります。



13.OKボタン

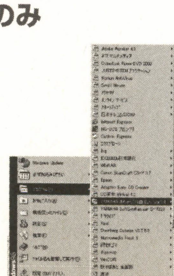
現在開かれている画面での作業が終了し，つぎへ進むときに使用します。

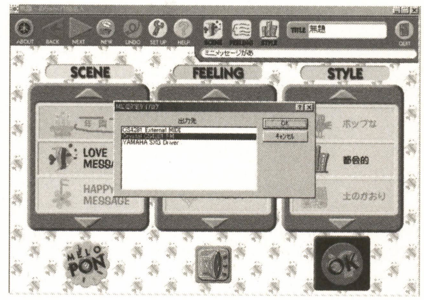


・バッキング画面：タイトル入力とメロポンのみ

では，一緒に曲を作っていきましょう！！

まず，画面左下の「スタート」→「プログラム」→「YAMAHA めちゃらく作曲名人」の順に選択していきます。ソフトが起動したら，バッキング画面が出てきます（その前に＜MIDI設定ダイアログ＞（出力先）という画面が出た人はYAMAHA SXG Driverを選択してください)。もしここで誤って選択しても，上のSET UPボタンを押せば同じ画面が出てきます。





最初にタイトルを記入するところが出てくるので，自分の好きなタイトルを入れてください（注意：曲の作成後でもタイトル変更可能なのでお気軽にね！)。

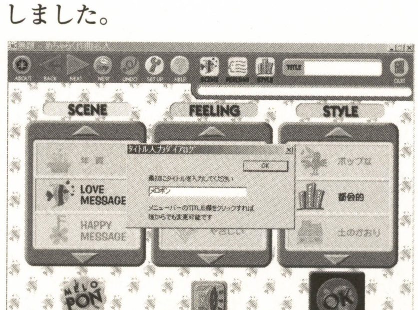


さて，タイトルを記入し終えたら，早速伴奏を作成しましょう！！ここではあなたの代わりにMELOPONボタンがSCENE/FEELING/STYLEの素材をスロット形式で自動構成してくれます。



ボタンを押し，スピーカで気に入るまでくり返し，納得いくものができたらOKボタンを押してつぎへ進みましょう！！

ここではタイトルを「メロポン」にしました。

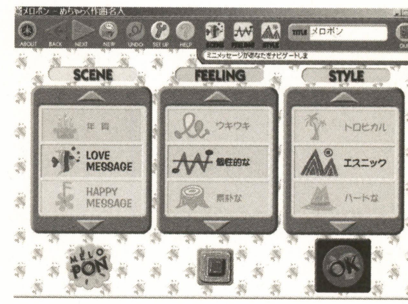


さて，タイトル記入が終わったら，早速伴奏を作成しよう！！ここではあなたの代わりにMELOPONボタンがSCENE/FEELING/STYLEの素材をスロット形式で自動構成してくれます。



ボタンを押し，スピーカで気に入るまでくり返し，納得いくものができたらOKボタンを押してつぎへ進みましょう!!

ここでは＜LOVEMESSAGE＞＜個性的な＞＜エスニック＞でOKしました。



・モチーフ入力画面：メロポンのみ

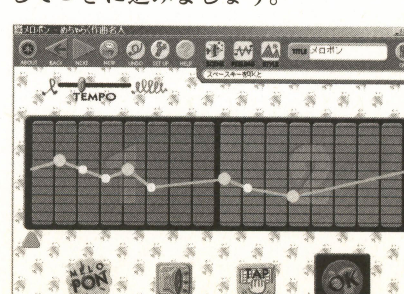
つぎに出てきたのは「モチーフ入力画面」，この画面は一つ前の「バッキング画面」で作った伴奏にメロディーのモチーフ※1を付けるんでしたよね。

それではモチーフを作ってみましょう。

まずは先ほど使ったMELOPONボタンとスピーカーボタンがこの画面にもあることを確認してください。ありましたか？「バッキング画面」と同じところにありますよね。そのMELOPONボタンを1回クリックしてみましょう。するとまん中の大きく1，2と書いてある表が折れ線グラフのように変化したと思います。このグラフは楽譜のようなもので，縦軸は音の高さを，横軸は小節分※2の時間を表していて，グラフ中のオレンジの点と白の点は音を出すポイントを表しています。このグラフを見れば音の高さとタイミングを直感的に知ることができます。

では今作成されたモチーフをスピーカボタンを押して聴いてみましょう。

どうですか？気に入ったモチーフはできましたか？気に入るまでこの手順を繰り返して気に入ればOKをクリックしてつぎに進みましょう。



※1　モチーフ：メロディの最小単位で主題とも呼ばれ，通常は2小節でできている。
※2　小節：楽譜の縦線と縦線で区切られた間の部分をさす。

・ソング構成画面：メロポンのみ

ようやく今回の曲作りも佳境に入ってきましたね，最後にこの「ソング構成画面」で「モチーフ入力画面」で作成したモチーフを基に曲全体を仕上げてみましょう。

まず画面のまん中より上にある1から32までの番号の付いた表をごらんください。この表は「モチーフ入力画面」の表と似ていて縦は音程，横は時間で曲全体を見渡せ，数字のついたところにはメロディーが入ります（二つ分でモチーフと同じ長さです)。

そしてメロディの数字四つ（4小節）の中には二つの点があると思います，この点は4小節ごとのメロディの起伏を表しています。

では早速ここでも必殺のMELOPONを使ってみましょう。画面右端にMELOPONボタン，スピーカボタン，OKボタンがあるのを確認していざ！

「ひっさ〜つ！」

「メ〜ロPON！！！」

すると表上のメロディの起伏が変化したと思います。スピーカボタンを押して曲全体を聴いてみましょう（ちなみに1小節目と2小節目は先ほどの「モチーフ入力画面」で作成したモチーフがそのまま使用されます)。ここでも自分のイメージに合うメロディーの起伏ができるまでMELOPONしてみましょう。



イメージに合った曲はできましたでしょうか？もう完成だと思ったら画面右上の「QUIT」ボタンをクリックしましょう。「現在の曲を保存しますか？」

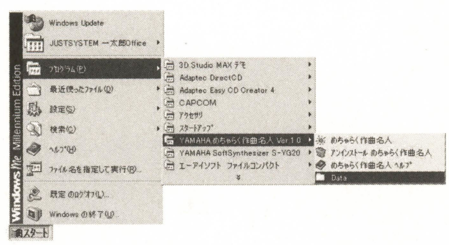


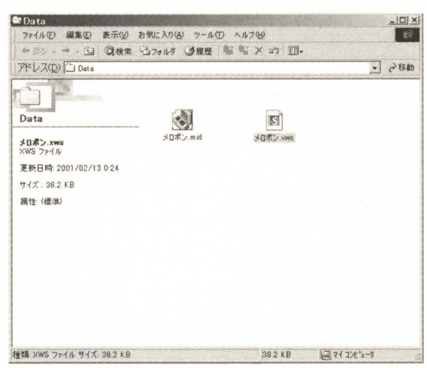
と出ますので「はい」をクリックして終了しましょう。もしEメールが使える環境であれば後に説明する「Eメールで曲を送る」にも挑戦してみましょう。

データの確認

先ほど作曲した曲がちゃんと保存されているか確認してみたいと思います。

作った曲は「スタート」→「プログラム」→「YAMAHA「めちゃらく作曲名人」Ver.1.0」→「DATA」の中に「曲タイトル.MID」と「曲タイトル.xws」という二つのファイル形式で保存されています。ありましたでしょうか？

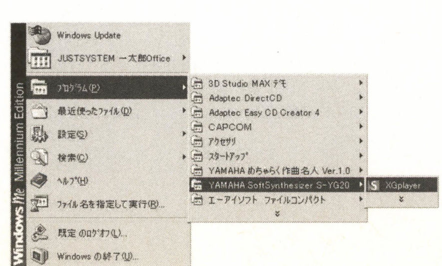




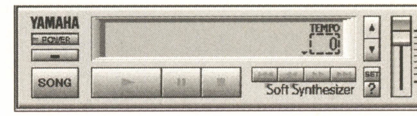
XGPlayerで再生

フトシンセサイザー「S-YG20」付属の「XGPlayer」で再生してみましょう。

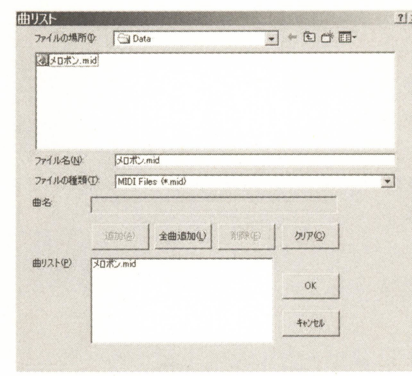
まず「スタート」→「プログラム」→「YAMAHASoftSynthesizerS-YG20」→「XGPlayer」の順で開いてください。



開きましたら左下の「SONG」をクリックしてみましょう。



「曲リスト」が開くはずです。「曲リスト」の上のほうの欄に曲名が出ていればその中の聴きたい曲名をダブルクリックしましょう，するとそれが下の欄に移ると思います。上の欄にそれらしい曲名が出ていない場合は「ファイルの場所」をもう一度確認してください（DATAフォルダのある場所は「ハードディスク内」→「ProgramFiles」→「YAMAHA」→「MRmeijin」→「DATA」です)。 聴きたい曲名が下の欄に移ったら「OK」をクリックします。



画面に曲名が出ましたでしょうか？出ましたら再生ボタンをクリックして自分の作曲した作品の鑑賞を心ゆくまで。お楽しみください。



Eメールで曲を送る

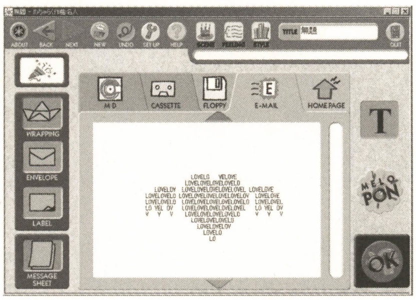
曲データが完成し，「OKボタン」を押したらつぎのような画面になります。



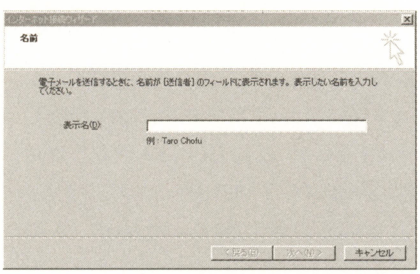
ここでは自作の曲をMDやカセットに録音したときに，そののジャケット

デザイン集などや，その自作の曲を好きな相手に送信することができます。ここでは，メールの送信の方法について解説します。

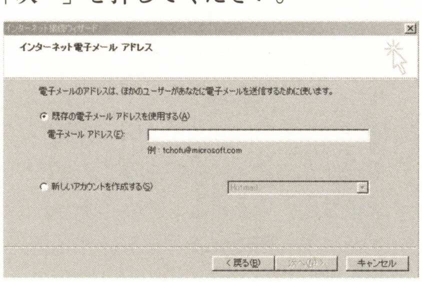
まず画面の「E-MAIL」のボタンを押してください。



ここに現れた画面は，相手に送ったメールの画面となります。ここでも「メロポンボタン」を使用することができます。画面が決まったら，つぎは「OKボタン」を押します。



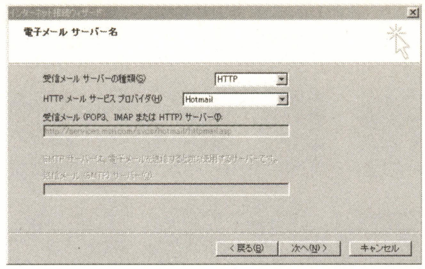
ここで入力する名前は，相手に届いたときに表示される名前ですので，自分の名前を書いてください。書いたら「次へ」を押してください。



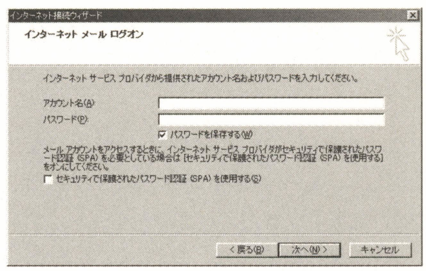
ここでは，既存の電子メールを書きましょう。

注意：ここへ書くメールアドレスは，プロバイダから取得したアドレスでもフリーメールでもかまいません。「新しいアカウントを作成する」を選択すると，hotmailの取得画面にアクセスされます。<hotmail：マイクロソフト社が提供するフリーメールで無料でアドレスが作れます>

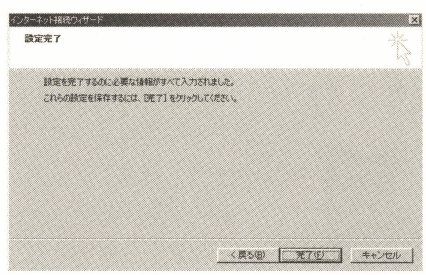
ここではhotmailを例にして進めます。アドレスを書き「次へ」を選択します。



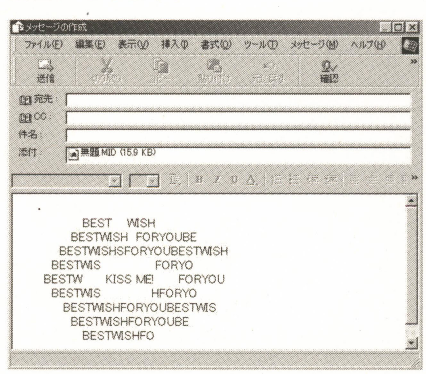
<電子メールサーバー名>が現れます。ここでも「次へ」を選択します。



ログイン画面が現れます。アカウント名はもう書かれていますが，パスワードには，自分が普段メールを送信するときに（プロバイダに接続するときに使用するパスワードを入れてください）すべて入れ終わり，「次へ」を押すと，設定完了がされましたので「完了」を押してください。



ここからは普段みなさんが見かける画面になります。



相手先のアドレスを書いて「送信」を押して送信完了です。

今回の復習

今回は「「めちゃらく作曲名人」」を使った作曲法の一つとして「MELO PON」を使った作曲方法について解説し，作曲した曲を「XGPlayer」で再生させ聴いてみるところまでやってみましたが，どうでしたか？作曲は簡単で楽しいものだと思っていただけましたでしょうか。お手本曲として今回誌面のほうで作成した「メロポン」という曲を弊社ホームページ上に掲載しておきますので参考にしてみてください。それとEメールができる人はぜひ友達同士で曲の交換などをして楽しんでみてください。

次回は今回紹介しきれなかった「めちゃらく作曲名人」のさまざまなエディットボタンを使っての作曲法を中心に解説していきたいと思います。

次回もおたのしみに！

＊次回予定

・「めちゃらく作曲名人」EDIT編
・インターネット活用法（未定）

「めちゃらく作曲名人」作品募集！

近々，「テーマ」と題して「めちゃらく作曲名人」の作品発表を弊社ホームページ上で開催しますので，皆さんお気軽に参加してみてください。初めてパソコンや作曲に挑戦された方，大歓迎です。参加希望の方は以下のデータをご用意のうえ，「めちゃらく作曲名人・作品募集」係まで郵送かまたはEメール（アドレス）に添付してお送りください。

簡単な曲でもかまいませんよ！では皆さんのご参加，お待ちしています。

・ご用意いただくもの

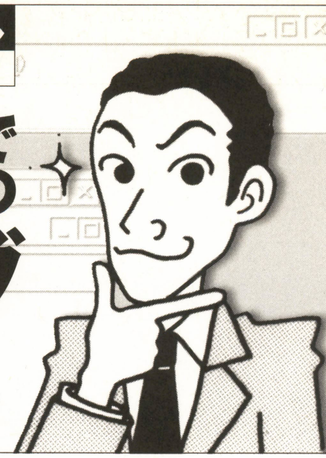
①1曲分のめちゃらく作品データ，MIDI・xwsファイル（郵送の方はフロッピーへ）
②作品の簡単な紹介，コメント
③作曲者プロフィール
④連絡先（Eメールアドレス）

作品募集テーマ

・「春」をテーマにしたメロポン作品発表会

TEXT & PROGRAMING：椛島たけし
協力：富士通南九州システムエンジニアリング

はじめよう Microsoft EXCELでの BASICプログラミング



新連載 第1回

「プログラミングは簡単にできる」

3月号まで「F-BASIC V6ではじめるWindowsプログラミング」でみなさんと一緒に勉強してきた椛島です。今月号からまた新たにMicrosoft EXCELを使ってプログラミングを行う講座の連載を始めることになりました。EXCELはあくまで道具でメインはBASICプログラミング!!。

わかりやすいものにしていきたいと思っています。よろしくお願いします。

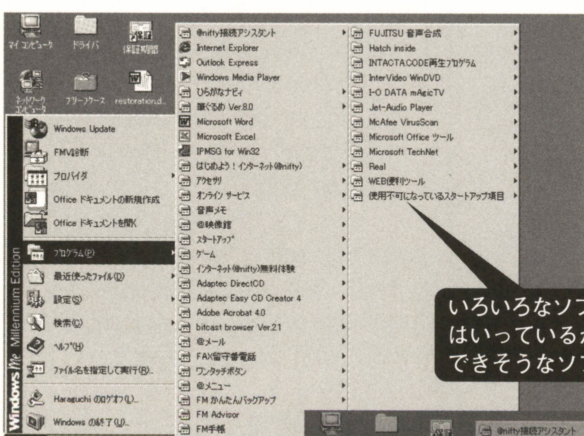
21世紀の現在はプログラミングの影が薄いのだ！

学校，会社，自宅，はてさてカバンの中と，身近にパソコンがある世の中になりました。ほんの数年前までは，なんだか特別な感じがしたインターネットも，ずいぶんと敷居が低くなった感があります。「買ったその日からインターネット」ってどこのコピーでしたっけ？

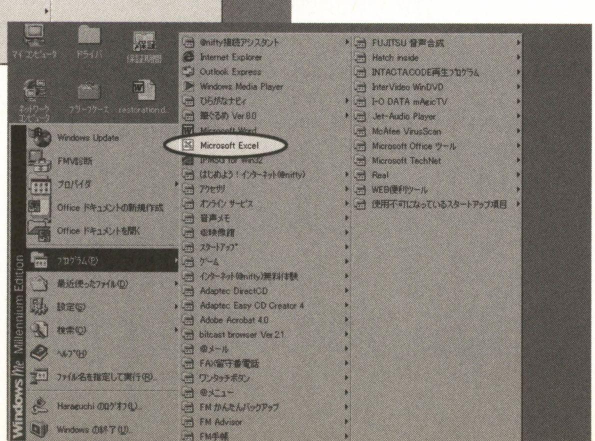
でも，それとは反対に「プログラム」や「プログラミング」はどんどん取っつき難いものになった気がします。

その昔は，パソコンやマイコンといえばプログラムでした。インターネットやゲームをするためにパソコンを買うのではなく，プログラミングをするために（プログラミング＝プログラムを作成する）パソコンやマイコンを買ったものです（注1）。

今は，プレインストール（パソコンに最初から組み込まれている）やバンドル（ついている）されているソフトに，一見プログラムが作成できるモノがない（第1図）ためか，パソコン初心者がプログラムに出会うことは少ないと思います。



《第1図》プログラムを作成できそうなソフトはあるかな？



《第2図》Microsoft Excelがあればプログラミングが可能だ

いますぐプログラムが作成できるぞ

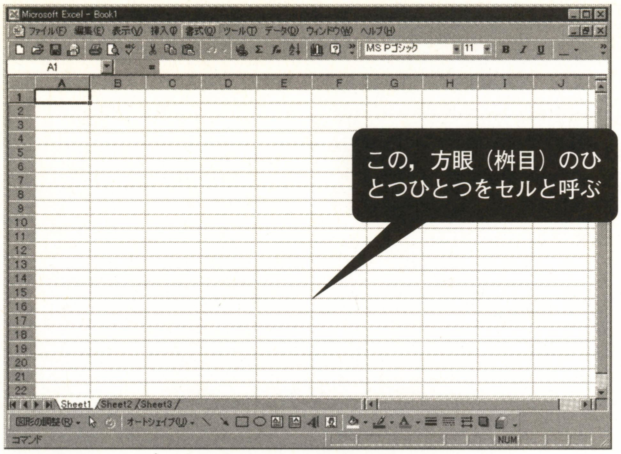
というわけで，「パソコンを持っているのにプログラミングしてない」という現状を憂いて登場したのがこの連載です。みんなの近くにあるパソコンには，たいてい『Microsoft EXCEL（以下EXCEL）』（第2図）がプレインストール（バンドル）されていると思います。なんと，このEXCELにはBASICなるプログラミング可能なツール（道具）がついているのですよ。

半年後には立派な？プログラマーだ

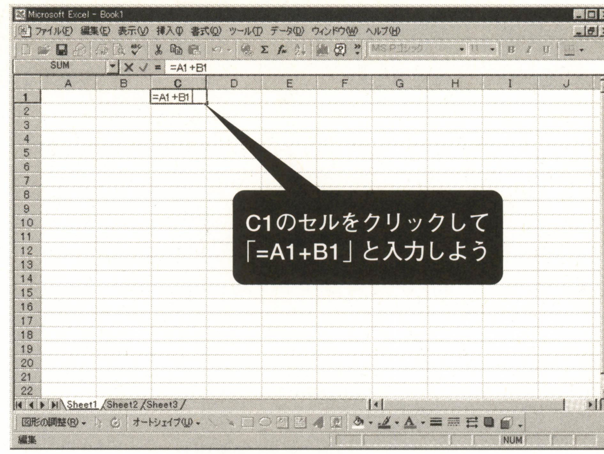
「プログラムが初めての人」「BASICなんて聞いたこともない人」もご心配なく。キーボードとマウスの基本的な操作ができる読者なら……きっと，半年後には立派なプログラムが作れるように，基本的な部分から解説していく予定です。そうですね，実用的でしかも遊び心があるプログラムを目標にがんばってみることにしましょう。

Microsoft EXCELって？

EXCEL（注2）とは表をもとに計算やグラフ作成などを行うソフトのことです。このようなソフトは，一般的に，表計算ソフトとかスプレッド・シートとかよばれています。まあ，とりあえず使って慣れてみることにしましょう。



《第３図》Microsoft EXCEL を起動したところ



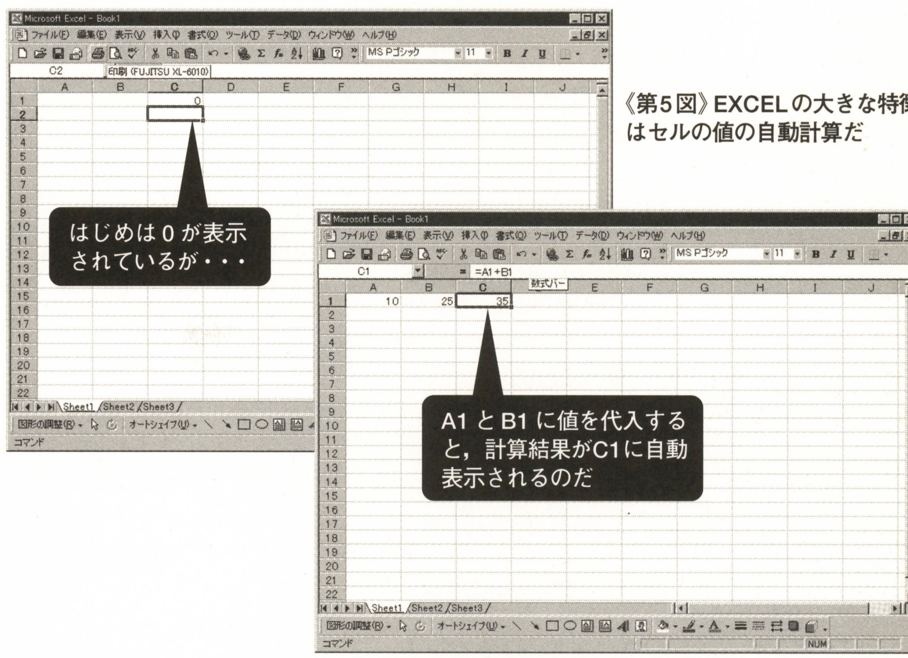
《第４図》とりあえずEXCEL に数式を入力してみると

Step1

まずはEXCEL を起動してみます（第3図）。まずは見た目ですが，いわゆる「方眼」がたくさんあるのが大きな特徴ですね。この方眼のことを「セル」（注3）と呼びます。

Step2

セル C の 1（今後 C1 と書きます）に「=A1+B2」と入力してください（第4図）。ただし，「」（カッコ）は入力する必要はありません。「+」の代わりに「*」でも「-」を入力しても結構です。とりあえず，C1 に 0（ゼロ）が表示されるはずです。ちなみに，たいていの場合，積算は×ではなく * を使用しますので注意してください。

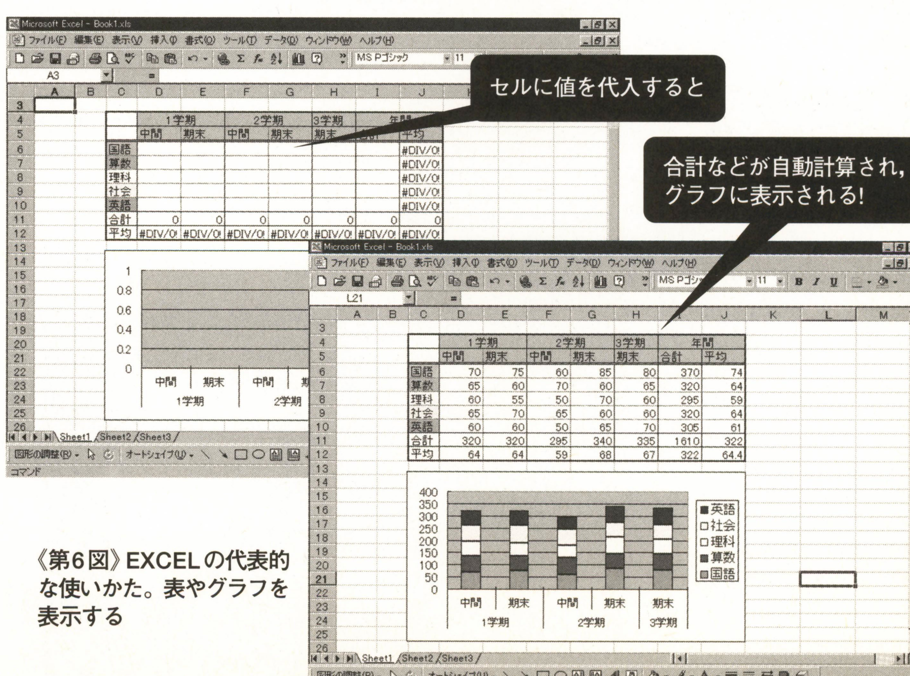


《第５図》EXCELの大きな特徴はセルの値の自動計算だ

Step3

セル"A1"と"B1"に好きな数字を入力すると・・・自動的にStep2 で入力した数式（計算）の結果を表示してくれるはずです（第５図）。

ここまでは，あたりまえといえばあたりまえの動作ですが，このような機能を応用させるといろいろなことが簡単にできるようになるわけです。EXCELの機能については，ちょっとした例（第6図）を紹介するだけにとどめておきます。例のように，EXCELを利用すると計算とか資料作成がとっても楽になることがわかると思います。ちょっと前まで，先生たちは電卓やソロバンでいちいち計算してたのに，今ではとっても楽になったというわけです。



《第６図》EXCELの代表的な使いかた。表やグラフを表示する

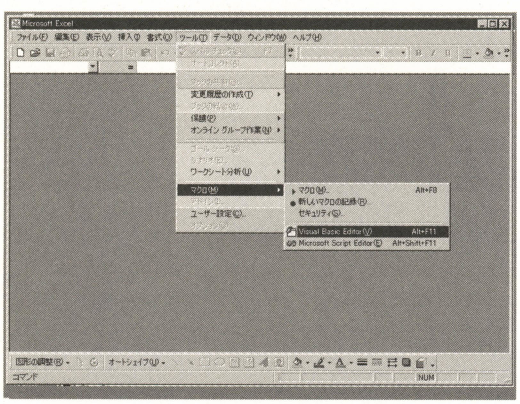
プログラムの基本は「Hello，World」

ここまでは，「EXCELの使いかた」であってBASICプログラミングではありません。この連載のテーマは「BASICでプログラミング」ということですので，そろそろプログラミングしてみることにしよう。

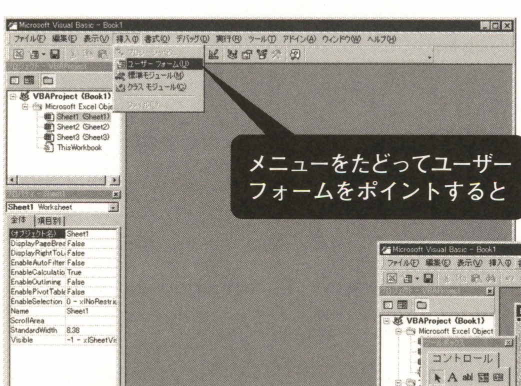
i-modeで話題のJAVAや，C言語，BASICでも，最初に覚えるのは「Hello, world」という文字を表示することが多いのです。では早速プロググミングにチャレンジしてみましょう。

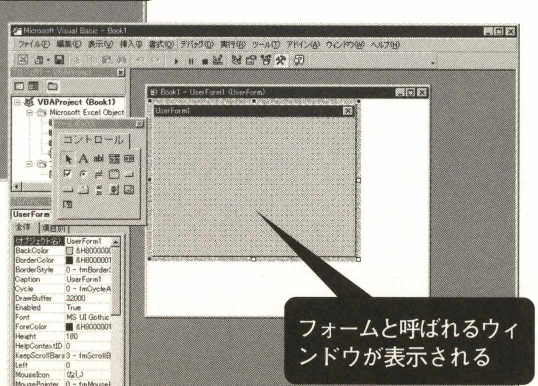
Step1

EXCELを起動し，メニューの[ツール][マクロ][Visual Basic Editor]を選択します（第7図）。



《第7図》EXCELでBASICを使うために，Visual Basic Editorを起動しよう





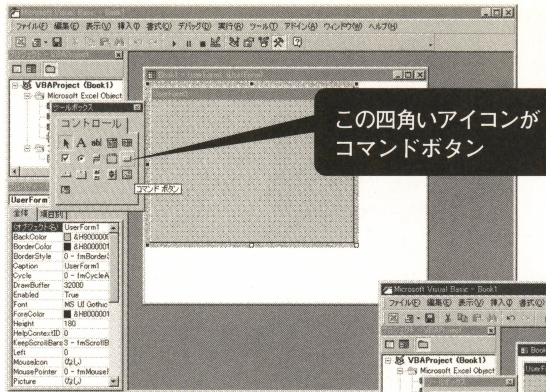
《第8図》Visual Basic Editorが起動されたら，画面デザインのためユーザーフォームを作ろう

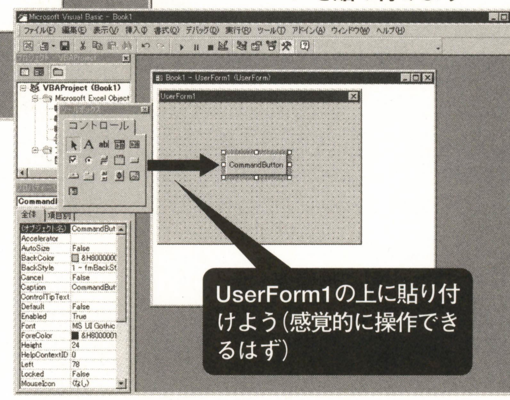
Step2

[挿入]メニューから[ユーザーフォーム]を選択します（第8図）。

Step3

ツールボックスのコマンドボタンをクリックしたあと，UserForm1をクリックし，コマンドボタンをUserForm1に貼り付けます（第9図）。





《第9図》ユーザーフォーム（UserForm1）が表示されたら，コマンドボタンを貼り付けよう

Step4

UserForm1のコマンドボタンをダブルクリック（もしくはコマンドボタンを右クリック後，[コードの表示]を選んでもOK）して，第10図ようなウィンドウを表示します。

そして，第11図のように
Private Sub CommandButton1_Click ()」と
End Sub
の間に，つぎの文字を入力します。

```
MsgBox "Hello,world"
```

Step5

[実行]メニューから[sub/ユーザフォームの実行]を選びます（メニューの下の三角のボタンをクリックしてもOK）。その後，表示されたウィンドウのコマンドボタンをクリックします（第12図）

どうですか？EXCELでもちゃんとプログラムができましたね。あんまり簡単だったので，拍子抜けしたでしょうか。もうちょっとプログラミングを続けることにしましょう。

Step6

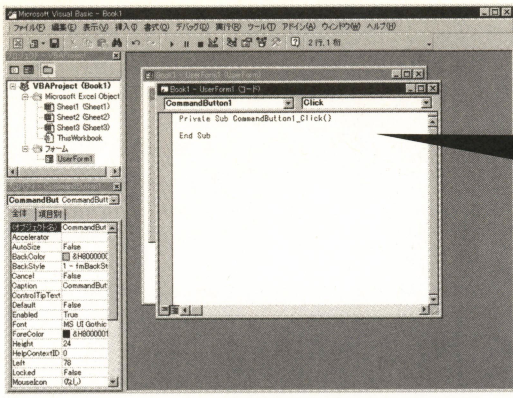
UserForm1の実行を終了し，さっき入力したところにつぎの文字を入力します（第13図）。

```
Dim a As Date

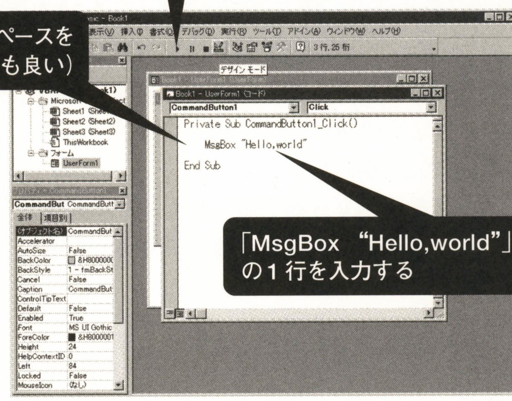
a = Time

If a < "12:00" Then
   MsgBox " 今は午前です "
Else
   MsgBox " 今は午後です "
End If
```

入力が終わったら，Step5の要領でプログラムを実行します（第14図）。



《第10図》コマンドボタンをダブルックリックすると，プログラム入力画面が表示される



《第11図》さっそく「コード」を入力しよう

まとめ

今回のポイントは，「プログラミングは身近に，簡単にできる」ということです。BASICの文法など，くわしい部分についてはあまり説明しませんでしたが，じょじょに説明していきますので安心してください。それでは来月をお楽しみに。

Technical Column

C言語のバイブルといわれる「プログラミング言語C」（B.W.カーニハン/D.M.リッチー著）以来，プログラミング入門は「Hello,world (hello, world)」という文字列をディスプレイ表示する例が非常に多い。もっとも，「プログラミング言語C」以前に「Hello,world」が使われていたかもしれないが，この本が広めることになったのはまちがいない。

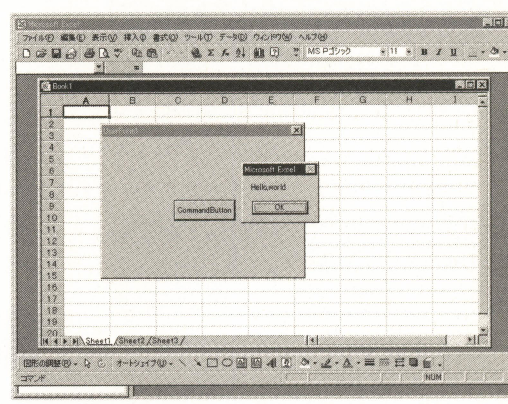
ちょっとだけ解説

BASICというプログラミング言語は，簡単な英単語を基本につくられているので，慣れさえすれば，誰でも簡単にプログラムができます。さきほど作成したプログラムも，解説なしでも，なんとなくわかったのではないでしょうか。とりあえず，Step6のプログラムを解説すると，

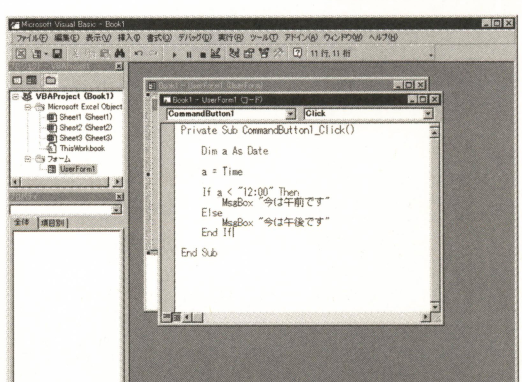
1.aという日付（時間）が入る変数を宣言します
2.aに現在時間を代入します
3.もしaに代入された時間が12:00より小さいなら「今は午前です」と表示します。そうではなかったら「今は午後です」と表示します。

ということです。あわせて，ステートメント（注4）の意味についても簡単に説明しておきます（今回はなんとなくわかればOKです）。

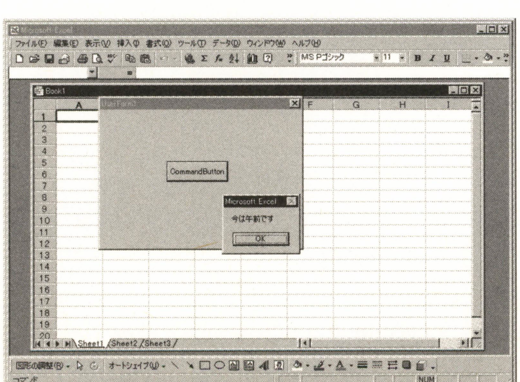
- MsgBoxステートメント……メッセージボックスを表示します
- Dim ステートメント……（指定された型の）変数を宣言します
- Timeステートメント……現在時刻を取り出します
- If ... Then ...Else ステートメント……条件にあった（あわない）処理を実行します



《第12図》コマンドボタンをクリック，メッセージボックスが表示される



《第13図》もう少しプログラムらしいコードを入力して実行してみよう



《第14図》午前と午後が正しく表示されれば成功！

（注1）ROMベーシックといって，OS（WindowsとかMS-DOS）が起動するまえに，BASICが起動するパソコンも多かったのだ。

（注2）本連載ではMicrosoft EXCEL 2000を使用して解説しています。しかし，Microsoft EXCEL 97でも大丈夫です。

（注3）セルとはCELL。つまり，小さな部屋や細胞のことです。

（注4）ステートメント　簡単に言うと命令のこと。

Linux環境でプログラミングを学ぼう！

よいこのLinux修行



第2回 シェルスクリプトを読んでみよう

横井 勝 <masaru@turbolinux.co.jp>

みなさんこんにちは，横井勝です。前回はインストールについて紹介しました。いきなりですが，この連載のタイトルに「プログラミング」という単語があるので，まずはプログラムを読むことにしましょう。今回は，シェルスクリプトという比較的簡単でありながら非常に奥の深い世界を紹介します。

Linuxをインストールしたばかりという方なら，ターミナルでシェルを利用することに慣れていないかもしれませんが，シェルスクリプトが理解できるとシェルの使い方も理解できるのではないかと思います。

1. シェルスクリプトとは

普段，ターミナルなどでユーザーと対話的に動作するシェルですが，プログラムの実行環境にすることもできます。ユーザーが実行したい内容をテキスト・ファイルにして，それをシェルに実行させることができるのです。シェルのプログラムは 「シェルスクリプト」（Shell Script）と呼ばれています。

有名なシェルとしては，Turbolinuxのデフォルトのシェルとされているbash（バッシュ），bashの原型となっているsh（エスエイチ），C言語に似た構文を書くことのできるcsh（シーエスエイチ），cshを発展させたtcsh（ティーシーエスエイチ），ksh（ケーエスエイチ）を発展させたzsh（ゼットエスエイチ）などがあります。これらのシェルは，シェルスクリプトの実行環境になります。

今回はbashのシェルスクリプトを読んでみます。シェルによってシェルスクリプトの文法が異なるので，違うシェルを使う場合はそのシェルのマニュアルを参考にしてください。

2. わからないことがあったらマニュアルを読みましょう。

マニュアルの読みかたがわかれば，シェルスクリプトなどにわからない使いかたが書かれていても，調べて理解することができるでしょう。ということで，まずはマニュアルの読みかたを紹介します。

わからないことがあったら，まずは何か資料を見て調べることが大事です。Linuxについても調べればわかることがたくさんあります。Linuxを使っていてわからないことがあれば，以下のような方法で調べることが可能です。

●ローカルの環境の資料を見る

- ○man コマンドを使ってマニュアルを表示する。
- ○info コマンドを使ってマニュアルを表示する。
- ○各種パッケージに収録されているマニュアルを見る。
- ○ソースを読む

●インターネットで調べる

今回はこの中でmanを使ったマニュアルの閲覧方法を紹介します。

2.1. コマンド man について

UNIX には伝統的に，各種ツール類のマニュアルを表示するための man というコマンドがあります。それでは試しに，ファイルの一覧を表示するコマンドのlsのマニュアルを見てみましょう。ターミナルのプロンプトに"man ls"と入力してみてください。《**第1図**》のような内容がでてくると思います。

いま，約720行ほどあるlsのマニュアルをターミナルで見やすいようにページャー（Pager）と呼ばれる種類のプログラムであるlessが，ターミナルの長さにあわせて表示を停止しています。スペース・キーかfキーを押すと続きが読めます。bのキーを押すと前のページに戻れます。

lessの使いかたのくわしい説明は "less --help" で表示される「コマンド概要」を見てください。

表示されているマニュアルの先頭のほうに「LS (1)」と表示されています。これは「第1章のls」ということを意味しています。マニュアルには八つの章があり，各章は以下のようになっています。

第1章 - ユーザーコマンド
一般的にユーザーが利用するコマンドのマニュアル

第2章 - システムコール
システムコール（カーネルに対する命令）についてのマニュアル

第3章 - ライブラリ関数
C言語などで利用するライブラリの関数を説明するマニュアル

第4章 - スペシャルファイル
/dev/ディレクトリに格納されているデバイスファイルなどのスペシャルファイルに関するマニュアル

第5章 - ファイルフォーマット
さまざまなプログラムが利用するファイルの書式を説明するマニュアル

第6章 - ゲームプログラム
ゲームのマニュアル

第7章 - 習慣やその他
システムを利用するための習慣や，文字集合の規格，ファイルシステムの構成など

第8章 - 管理コマンドと特権コマンド
スーパーユーザーが利用するコマンドのマニュアル

マニュアルによっては，複数の章に同じ名前のマニュアルがあります。たとえば，printfというマニュアルは第1章と第3章にあります。たとえば，C言語のプログラムを書いているときにprintf()の関数を調べたい場合は"man 3 printf"と章の番号を指定してください。

2.2. うわっ，マニュアルが英語だぁ。

マニュアルによっては，英語のマニュアルが表示される場合があります。Linuxのソフトウエアのほとんどは英語のマニュアルが収録されていますが，日本語のマニュアルは収録されていません。翻訳されていないマニュアルは，原文である英語のマニュアルが表示されてしまうのです。

Linuxの日本語のマニュアルのほとんどは，英語で書かれていたマニュアルを多くの有志によって日本語に翻訳されています。これら日本語のマニュアルは，JM Project（http://www.linux.or.jp/JM/）やX Japanese Documentation Project（http://xjman.dsl.gr.jp/）が，翻訳されたマニュアルを管理して公開しています。

英語のマニュアルが表示されたからといって挫けないでください。高校で習う英文法がわかっていれば，専門用語などは辞書で調べながら読める内容です。英文の読解力もつくので，がんばって読んでみてください。

2.3. いろんなコマンドを調べてみよぉ。

ということで，ファイルの一覧表示を行うls，ファイルのコピーを行うcp，ファイルの移動を行うmv，ファイルの削除を行うrmのマニュアルを，manで調べてみてください。これらの基本的なコマンドには，普段はあまり使わないような機能を含め，たくさんの機能があります。無理して全てのオプションを暗記する必要はありません。必要な時に必要なオプションを覚えていけばいいでしょう。

```
LS (1)                                                    LS (1)

名前
        ls, dir, vdir - ディレクトリの中身をリスト表示する

書式
        ls [オプション] [file...]

        POSIX オプション [-CFRacdilqrtu1]

        GNU  オプション (簡略形式) : [-1abcdfghiklmnopqrstuvxABCD-
        FGHLNQRSUX]  [-w  cols]  [-T  cols]  [-I  pattern]
        [--block-size=SIZE]                      [--classify]
        [--color[={none,always,auto}]] [--file-type] [--full-time]
        [--format={across,commas,long,single-column,verbose,verti-
        cal}] [--human-readable]  [--indicator-style={none,file-
        type,classify}]            [--quoting-style={c,escape,lit-
        eral,shell,shell-always}] [--show-control-chars]  [--si]
        [--sort={none,extension,size,time,version}]
        [--time={atime,access,ctime,status,use}] [--help]  [--ver-
        sion] [--]

:
```
（脚注1 -- POSIX）

《第1図》"mn ls"をしたとき

（脚注1）POSIX（ポジックス：Portable Operating System Interface for UNIX）…IEEE 1003.1で規定されているUNIXの標準アプリケーションインターフェースの仕様です。UNIXをベースにした複数のOSが開発され，それぞれ独自の拡張がされました。これら複数のOSでのアプリケーションの互換性を実現するために，定義されたのがPOSIXです。

3. シェルスクリプトの現場から…

前置きが長くなりました。それでは実際に活躍しているシェルスクリプトを読んで，シェルスクリプトの代表的な使いかたを修行しましょう。

誰もが絶対にお世話になるシェルスクリプトは，システム起動時に実行されるシェルスクリプトでしょう。Linuxでは，システム起動時に/sbin/initというプログラムが/etc/inittaという設定ファイルで設定されたプログラムを実行します。/etc/inittab の中の「si::sysinit:/etc/rc.d/rc.sysinit」と書かれた部分が 起動時に実行するプログラムを指定しています。この設定では，起動時に/etc/rc.d/rc.sysinitを実行することになります。inittabのくわしいことはman inittabで調べてください。

それでは/etc/rc.d/rc.sysinitを先ほど紹介したページャであるlessを使って見てみましょう。シェルで "less/etc/rc.d/rc.sysinit"とタイプするとlessが/etc/rc.d/rc.sysinitを表示してくれます。

このシェルスクリプトは構造としては難しくありません。しかし，システムの起動時の設定などを行わなければならないため，やっていることは難しいことばかりです。各々の部分の目的は今回は紹介できません。知りたい方は，このシェルスクリプトから実行されるプログラムをmanで調べてみてください。 いろいろなことがわかると思います。

3.1 1行目の約束

```
1:   #! /bin/sh
```

1行目には「#!/bin/sh」と書かれています。この「#!」の後ろに書かれているのは，このスクリプトを実行するプログラムで，「/bin/shでこのシェルスクリプトを実行する」という意味になります。たとえば，Perl（パール）のスクリプトの場合，「#!/usr/bin/perl」と指定することになります。

本来，shのシェルスクリプトは"sh[ファイル名]"という具合に実行するのですが，ここの1行目のような「#!」で実行するプログラムを指定したうえに，このファイルに実行権限をつけると("chmod +x [ファイル名]" をすると)，そのファイルが実行ファイルとなり，普通のプログラムのように実行できるという約束があるのです。

Turbolinux の場合，/bin/shは /bin/bash へのシンボリックリンク（**→脚注2**）となっています。実際に利用するのはbashになります。それでは，bashのマニュアルを参考にしながら /etc/rc.d/rc.sysinit を読んでいきましょう。

3.2 コメントを書くには

```
2:   #
3:   # /etc/rc.d/rc.sysinit - run once
     at boot time
4:   #
5:   # Taken in part from Miquel van
     Smoorenburg's bcheckrc.
6:   #
```

そのあとの数行は「#」が行頭についています。シェルスクリプトの場合，この「#」から行末（改行）まではコメントという扱いになり，処理しません。

実際にシェルスクリプトを書くときに，書いたあとに理解できるようにコメントを書いたり，一時的に動かしたくない部分などを動かさないようにするときに「#」を活用してください。

3.3. 変数と環境変数

```
9:  PATH=/bin:/sbin:/usr/bin:/usr/
    sbin
10: export PATH
```

ここでは，変数"PATH"に「/bin:/sbin:/usr/bin:/usr/sbin」という内容を代入して，「export PATH」で変数PATHを環境変数として，このあとに実行するプログラムに，この変数を渡すように指定しています。

変数の場合は，そのシェルスクリプトの中でのみ利用可能ですが，環境変数にすると，そのシェルスクリプトから実行されるプログラムへ変数が渡されます。そしてシェルの場合，環境変数をそのシェルでの変数として扱います。

ここに出ている環境変数PATHには特殊な意味があります。たとえば，lsというコマンドは/bin/lsというファイルになります。シェルでlsとタイプして/bin/lsが実行されるのは，PATH に/bin が指定されているためです。もし環境変数PATHに/binが設定されていない場合，lsを実行するには"/bin/ls"と指定しなければなりません。シェルでは，PATHに指定されているディレクトリからプログラムを探すようになっているのです。

PATHには実行するプログラムがあるディレクトリをコロン「:」で区切って複数指定できます。ここで指定されている「/bin:/sbin:/usr/bin:/usr/sbin」の場合，/bin,/sbin,/usr/bin,/usr/sbin の順にコマンドを探すようになります。

現在使っているシェルのPATHを見るには"echo $PATH"とタイプしてみてください。変数PATHの中身が表示されます。

3.4. if 文

条件によって動作を変えるために，シェルにはif文があります。まずはif文の文法を紹介します。たとえば，以下のような式があるとします。

```
if <テスト 1>
    then <やること 1>
    elif <テスト 2>
    then <やること 2>
    else <やること 3>
fi
```

「テスト1」を実行して「テスト1」が0で終了する正常な終了の場合，「やること1」を実行します。そうでなく「テスト1」が0以外で終了した場合，「テスト 2」を実行して，「テスト 2」が 0 で終了した場合，「やること 2」を実行します。「テスト 2」も0以外で終了した場合，「やること 3」を実行します。

elifとelseの部分は必要なときに使ってください。「if [テスト] ; then [やること] ; fi」と書くこともできます。

それでは，実際に利用している部分を見てみましょう。

```
12: # Read in config data.
13: if [ -f /etc/sysconfig/network ];
    then
14: . /etc/sysconfig/network
15: else
16: NETWORKING=no
17: HOSTNAME=localhost
18: fi
```

ちなみに，ここにある「;」は，テキストで改行をしているのと同じ意味になります。

13行目のifのあとの括弧「[」はコマンドになっていて，testというコマンドへのシンボリックになっています。「[」の使いかたは「man test」で調べてください。マニュアルを読むと「[-f /etc/sysconfig/network]」という部分は「普通のファイル /etc/sysconfig/network が存在すれば真になる」というふうに理解できます。

本来はコマンドであるはずの「[」ですが，bashやzshではこのコマンドはシェル本体に組み込まれたビルトインの機能となっています。

14 行目の「. /etc/sysconfig/net

（脚注2）シンボリックリンク（Symbolic link）…本来あるファイルの名前以外に別の名前でもそのファイルを利用できるようにするための特殊なファイルです。たとえば/bin/shは/bin/bashへのシンボリックリンクとなっているため，/bin/shの読み書き実行をすると/bin/bash読み書き実行をすることになります。似たファイルに「ハードリンク」というものがあります。くわしくはman lnで表示されるマニュアルをご覧ください。

work」のドット(".")はsourceというふうに読みかえてください。sourceは，指定されたファイルの中身を実行しているシェルスクリプトで評価するという意味になります。

ここのif文では「普通のファイル /etc/sysconfig/networkが存在するなら，/etc/sysconfig/networkの中身をこのシェルの中で実行し，/etc/sysconfig/network が存在しない場合は変数NETWORKING を"no"に，変数HOST NAMEを"localhost"にする」という動作をします。

3.5. 他のコマンドの実行結果を利用する

それでは，ちょっと読み飛ばしてgrep(グレップ)の実行結果をif文の条件に利用する例を紹介します。

まずは，grepの使いかたを紹介します。grepは，探したい文字列を探したいファイルから探すときに使うと便利なコマンドです。たとえば"abc"という文字列を含んだ行を三つのファイル file-1, file-2, file-3 から表示させる場合，"grep abc file-1 file-2 file-3"と実行すると，文字列"abc"を含んだ行を表示してくれます。grepにもたくさんのオプションがあるので，くわしくはman grepを見てください。

それではスクリプトを見てみましょう。

```
77: if grep -i nopnp /proc/cmdline
      >/dev/null ; then
78:     PNP=
79: else
80:     PNP=yes
81: fi
```

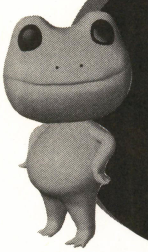
77行目の「grep -i nopnp /proc/cmdline >/dev/null」という部分で，grepを用いています。うしろの「>/dev/null」という部分では，grepが出力する文字列を/dev/nullというファイルに書き込んでいます。「>」は，その標準出力(stdout)を指定されたファイルに書き込むという意味です。標準出力についてはman stdoutを参照してください。「>>」とすると，指定されたファイルに追加書き込みを行います。「¦ (パイプ)」の後ろにコマンドを指定すると，標準出力をパイプの後ろに指定されたコマンドの標準入力(stdin)に渡されます。「>」,「>>」,「¦」の使いかたについては，man bashに載っています。/dev/nullはスペシャルファイルで，最終的には標準出力を捨てることになります。/dev/nullについてはman null で調べれます。

それでは，grepで何をしているのか調べてみましょう。ということで，manを使ってgrepのマニュアルを見てみます。grepは「grep [オプション] <パターン> [ファイル]」という形式で実行して，パターンに一致する行を表示するそうです。パターンに一致した場合は終了時に0を返して終了して，パターンに一致しない場合は，終了時に1を返して終了する，ということがマニュアルには書かれています。ここでは grep のオプションに"-i"を指定しています。このgrepのオプション-iをmanで調べると，"--ignore-case"を指定するのと同じで，「パターンと入力ファイルの双方で英大文字と小文字の区別をしない」と説明されています。この5行では，「/proc/cmdline に，大文字，小文字を区別しないでnopnpという文字がある場合，変数PNPを空に，文字が見つからない場合は変数PNPにyesという文字列を代入しています。

4. おわりに

シェルでは，多くのコマンドを上手に組み合わせて，便利なプログラムを書くことができます。それら多くのコマンドに関しては説明できませんが，マニュアルを読めば，さまざまなことがわかると思います。多くのコマンドの使いかたを調べてみてください。きっとLinuxが楽しく使えます。

プログラマーをめざす人の必見講座

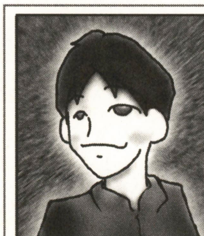


図解！ Visual Basic ゲーム作成術

第2回　キャラクター同士の衝突判定はどうプログラミングする？

ゲームに複数のキャラクターが登場する場合，キャラクターの移動範囲が重なっているとき「キャラクター同士が衝突する」可能性があります。この衝突が，ゲームの進行に影響しない場合は，何も処理を行う必要はありません。しかし，シューティング・ゲームでの弾のように，衝突することがポイントになる場合は“衝突判定”を行う必要があります。今回は，この衝突判定を行って，キャラクター同士が干渉し合う簡単なゲームを『Microsoft Visual Basic（以下VBと略）』で作ってみましょう。

TEXT&PROGRAMMING●東　幸太



BASIC歴15年以上のプログラマー。プログラミング可能な携帯電話の登場で，趣味プログラミングの領域がまた広がった。現在の機種は，実行速度やメモリ空間の点で制約が多いとはいえ，可能性の面では興味はつきない。将来，実行速度などが改善されていけば，楽しい装置になっていきそうだ。

Microsoft Visual Basic Programming System for Windows Ver.6.0（日本語版）

オープン価格

※編集部の価格調査によると，下記価格が目安になります。

ラーニング版2万円強
（アカデミック・パッケージ1万円弱）
プロフェッショナル版8～9万円
（アカデミック・パッケージ1万円強）

問い合わせ先
マイクロソフト
☎03-5454-2300

ホームページ
http://www.microsoft.com/japan/

動作環境
Windows 95／98／2000／NT4.0が動作するマシン



キャラクター同士の関係を理解しよう

一般的なゲームは，登場するキャラクターが複数いますよね。複数いる場合は，それぞれのキャラクター同士の関係に対応したプログラムを組み込むことが必要になります。

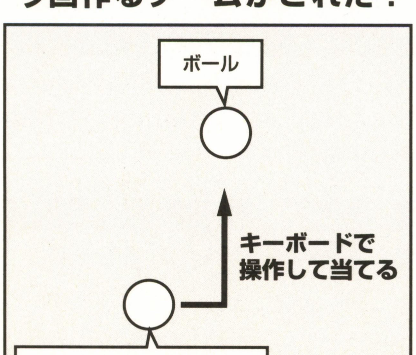
前回作成した『ライン・トレース・ゲーム』のように，一つのキャラクターしか登場しない場合は，このような処理を考える必要はありません。しかし，二つ以上のキャラクターが登場する場合は，「それぞれのキャラクター同士がどのような関係にあるか」をはっきりさせておく必要があります。そして，その関係に応じて，プログラムに衝突判定を組み込み，衝突したときの処理を追加していかなければなりません。

たとえば，**第1図**のような簡単なゲームのケースで考えてみましょう。このゲームには，プレイヤーが操作するプレイヤー・キャラクターだけでなく，目標である“ボール”というキャラクターが登場します。プレイヤー・キャラクターがボールに体当たりして，ボールを吸収していくと得点が増えることにします。

このゲームの場合，登場するキャラクターは2種類になります。それぞれのキャラクターが同じゲーム・フィールド上に存在するようになると，それぞれのキャラクター同士の衝突が起こるようになります。衝突したときに何も処理を行わなければ，キャラクター同士は何もなかったかのようにすれちがうことになります。しかし，プレイヤー・キャラクターとボールが衝突した場合は，「吸収して得点を増やす」という処理を行わなければなりません。

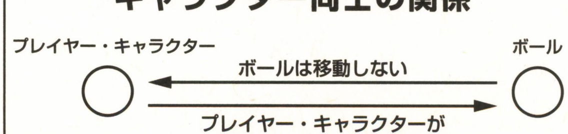
このような関係を図に示すと，**第2図**のようになります。この図に対応するように，プログラムを組んでやればよいことになります。

今回作るゲームがこれだ！



《第1図》衝突判定を考えるために，このようなシンプルなゲームを作ってみる

キャラクター同士の関係



《第2図》登場するキャラクターに対して，おたがいの関係をはっきりさせておいてからプログラミングしよう

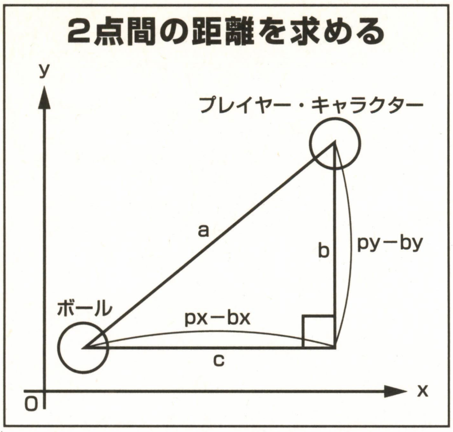
いろんな衝突判定の方法がある！

衝突しているかどうかを判断する方法としては，代表的なものとして

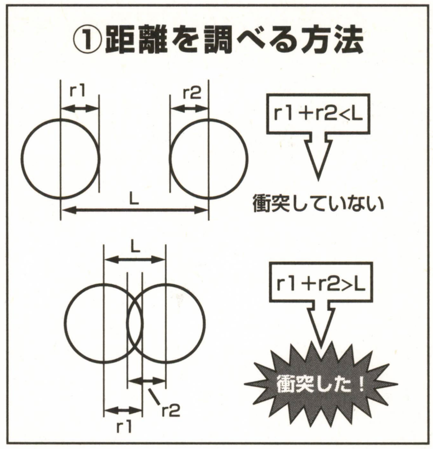
①距離を調べる方法
②座標の比較を行う方法

などがあります。

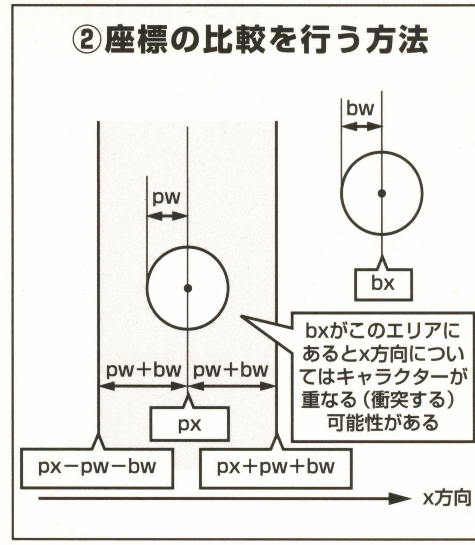
①の距離を調べる方法では，キャラクター同士の座標からおたがいの距離を計算し，衝突するほど近くに存在



《第3図》三平方の定理から2点間の距離を求めて，距離の値を見て衝突判定を行う



《第4図》「r1＋r2」とLを比較して，Lのほうが大きければ衝突しており，Lのほうが小さければ衝突していないことになる



《第5図》x方向について衝突の可能性あるなら，その後y方向についても調べ，重なった場合は衝突と判定する

するかどうかを調べます。

プレイヤー・キャラクターの座標を（px，py），ボールの座標を（bx，by）とすると，二つのキャラクターの距離Lは以下のような式で計算できます。

●数式

$L=\sqrt{(px-bx)^2+(py-by)^2}$

●BASIC言語表記

```
L=sqr((px-bx)^2+(py-by)^2)
```

この式は，数学の「三平方の定理」を利用したものです。簡単に解説しましょう。

第3図のように，プレイヤー・キャラクターの座標とボールの座標を頂点に持つ，直角三角形を想定します。**第3図**のような直角三角形の辺の長さには，

$a^2=b^2+c^2$

という関係がつねに成立します。この関係に，実際の座標をあてはめてみると，

$L^2=(px-bx)^2+(py-by)^2$

となります。両辺の平方根を計算すると，最初の数式になるわけです。二つのキャラクターの位置によっては，「px－bx」や「py－by」がマイナスの値になる場合もありますが，2乗している段階で符号がうち消されるため，問題なく計算できます。

三平方の定理を習っていない人は，「この式を書いておけば変数Lに2点間の距離が入る」と理解してください。

距離が計算できたら，キャラクターの大きさを考慮して，衝突しているかどうかを判断します。**第1図**のゲームの場合，キャラクターはどちらも円状になっていますね。ですから**第4図**のように，キャラクターの円の半径を合計した値よりも距離が小さい場合には衝突，そうでない場合は衝突していない，ということになります。

ゲームに応じて衝突判定方法を選択する

つぎに，②の座標の比較を行う方法を紹介しましょう。これは，キャラクター同士のX座標とY座標をそれぞれ比較し，キャラクター同士が重なっているかどうかを判断するというものです。

処理の流れとして，まずX座標について，衝突している可能性があるかどうかを判断します。プレイヤー・キャラクターの幅の1／2をpw，ボールの幅の1／2をbwとすると，

```
if px + pw + bw > bx and
px - pw - bw < bx then …
```

という条件判断を行えば，**第5図**のようにX方向についてキャラクターが重なる可能性があるかどうかが判断できます。

この条件が成立した場合は，Y座標についても同様に判断を行います。プレイヤー・キャラクターの高さの1／2をph，ボールの高さの1／2をbhとすると，

```
if py + ph + bh > by and
py - ph - bh < by then …
```

という条件判断で，Y方向についてキャラクターが重なる可能性があるかどうかを判断できます。X方向，Y方向のどちらについても重なる可能性がある場合は，衝突していると判断します。

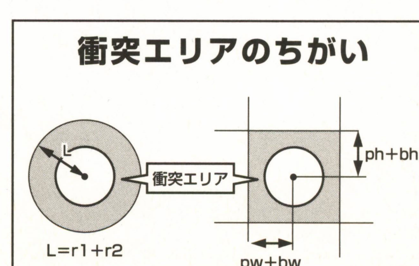
距離を調べる方法では，衝突判定におけるキャラクターの形が円になりますが，座標の比較を行う方法では長方形になります。このため，ゲーム・デザインに応じて，適切な方法を選択する必要があります（**第6図**）。

たとえば，キャラクターの形が円に近い場合は距離で判断するほうがよいですし，棒状のキャラクターや長方形のキャラクターの場合は座標比較のほうが有利です。

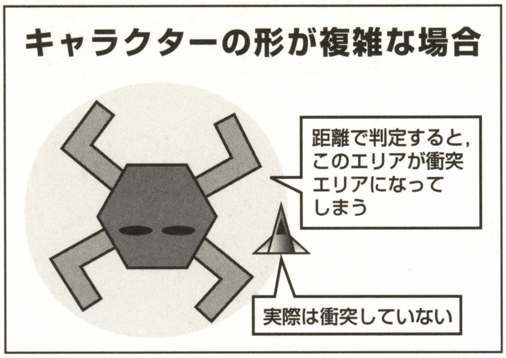
キャラクターが複雑な形状の場合はどうする？

第1図のゲームでは，衝突判定対象となるキャラクターがどちらも円になっているため，距離を計算するだけで厳密な衝突判定が行えます。しかし，通常はキャラクターの形が複雑ですから，距離だけで判断すると，**第7図**のような場合は衝突判定が正しく行えません。

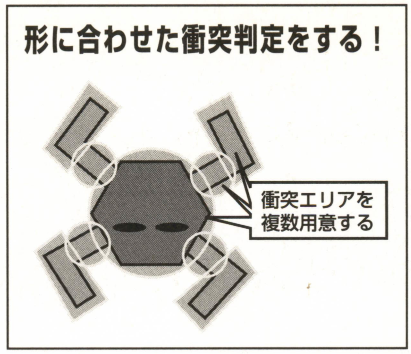
複雑なキャラクター同士の衝突判定を厳密に行うためには，多くの処理時間が必要となります。それほど速度が期待で



《第6図》①距離判定の場合は円形，②座標判定の場合は長方形になる



《第7図》プレイヤーに不利な判定を行うと，ストレスを感じさせてしまうので注意しよう



《第8図》キャラクターの形に合わせて，衝突エリアを複数持たせればいい

きないVBでは，実行できない場合もでてくるでしょう。そこで，衝突判定を簡略化してやる必要があります。

たとえば**第7図**のような場合は，**第8図**のように「複数か所で衝突判定を行う」という方法があります。この場合，衝突判定におけるキャラクターの形は**第8図**のようになるため，もとのキャラクターの形とは一致しません。しかし，動きのあるゲームであれば，この程度に簡略化してしまってもそれほど違和感は生じないものです。

ここで注意しなければならないのは，「簡略化の際にはプレイヤーが有利になるように簡略化してやる」という点です。たとえば，シューティング・ゲームの場合，プレイヤーの弾が敵キャラクターに衝突しているかどうかという判断は，なるべく衝突しやすく判断します。逆に，プレイヤー・キャラクターと敵キャラクターの衝突判定は，少々接触していても衝突していないように判断してやります。

簡略化の結果，見た目以上にプレイヤー・キャラクターが敵キャラクターに衝突されやすくなっていたりすると，ゲームプレイ時にプレイヤーにストレスがかかってしまいます。これでは，「おもしろいゲームだけどプレーしたくない…」と思わせてしまいます。

また，あまりにもプレイヤー有利に設定すると，ゲームが簡単になりすぎてゲーム自体が成立しなくなってしまいます。しかし，基本的な方針としては，プレイヤーに甘く設定しておくのが一般的です。

サンプル・ゲームで衝突判定を体験しよう！

それでは，実際にVBで衝突判定を含んだゲームを実行してみましょう。サンプルとして，**リスト1**の『ボール・キャプチャ・ゲーム』を使います。

リストを入力して実行すると，**第9図**のような画面が表示されます。画面中央には，プレイヤーが操作するプレイヤー・キャラクターが白い円として表示されます。また，ボール（赤い円）が，ゲーム・フィールドのどこかに表示されます。

プレイヤーは，プレイヤー・キャラクターをテンキーの[2][4][6][8]キーで移動させ，ボールに体当たりさせることでボールを回収できます。ボールが回収されると得点が増え，別の場所に新たなボールが表示されます。制限時間内に，なるべく多くのボールを回収してください。

リストで衝突判定がどう使われている？

このゲームでは，プレイヤー・キャラクターに加えてボールが登場します。両者が接触した場合には得点を増やし，新しい位置へボールを設定する，という処理を行う必要があります。このために，プレイヤー・キャラクターとボールの衝突判定を行っています。

両方のキャラクターが円に簡略化されているため，衝突判定は中心座標同士の距離を求める方法が適切です。関数として，distanceというものを定義し，この関数で距離を求めて衝突判定を行っています。

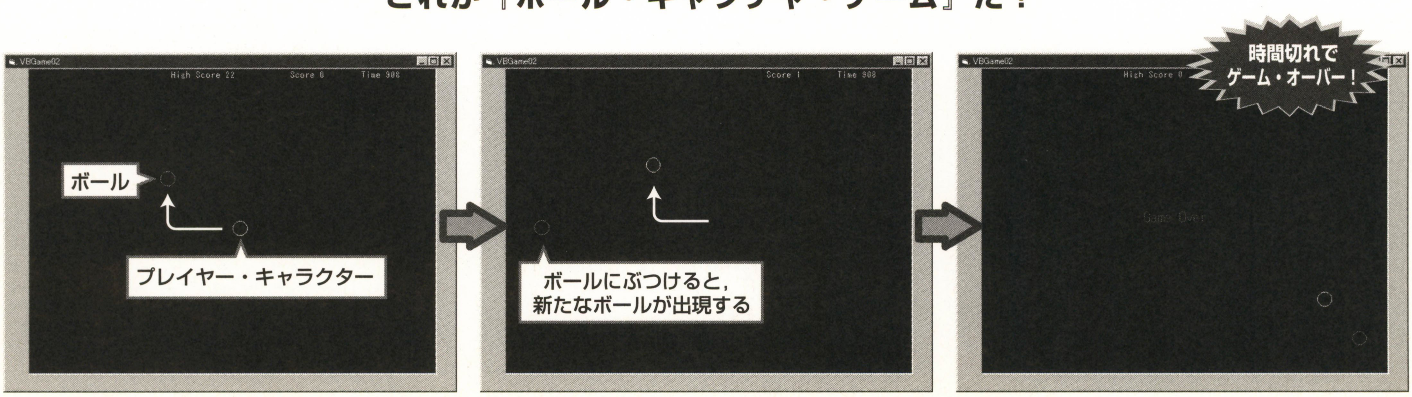
関数distanceは，

```
L=distance(x0, y0, x1, y1)
```

と呼びだすことで，座標点（x0，y0）と（x1，y1）の間の距離を計算し，返値とします。よって，この場合は変数Lに距離が格納されるわけです。

衝突判定を行うのタイミングは，どちらかのキャラクターが移動したり，キャラクターの外形が変化したりして，衝突判定を行う範囲が変化したときになります。このゲームでは，移動するキャラクターはプレイヤー・キャラクターだけですし，変形することもありません。よって，プレイヤー・キャラクターの移動処理が終わり，新しい位置がきまったとき

これが『ボール・キャプチャ・ゲーム』だ！



《第9図》ボールを取るとき，余計な動きを加えてしまうと時間のムダになり，得点は伸びなくなる。ボール出現パターンを記憶して，最短距離を移動しよう

に衝突判定を行っています。

ボールの出現パターンは“乱数（規則性や繰り返しのないでたらめな数のこと）”で決定しています。乱数の初期値を毎回一定にしているため，ボール・キャラクターの登場するところは，毎回同じものになります。乱数の初期値は，「rnd（−1）」を実行してからrandomizeを実行することで，同じものになります。まったくランダムに登場するところをきめると純粋な反射神経のゲームになりますし，パターンをきめておくとパターン記憶と先読みのゲームになりますよ。

まとめ

ほとんどのゲームには，複数種類のキャラクターが登場するため，キャラクター同士の衝突判断が欠かせません。

衝突判断の基本は距離の計算ですが，キャラクターの形状が複雑な場合はいろいろな簡略化が必要となってきます。簡略化の方法次第で，必要十分な判断を高速に行うことも可能です。ゲーム・デザインに応じて，うまく簡略化した衝突判断を組み込んでください。

《第1表》『ボール・キャプチャー・ゲーム』の変数

変数名	内容
hiscore	ハイスコア
score	プレイヤー・スコア
playerx	プレイヤー・キャラクターX座標
playery	プレイヤー・キャラクターY座標
ballx	ボールX座標
bally	ボールY座標
ti	残り時間
gamestat	ゲーム進行状況（0：ゲーム中，1：ゲーム・オーバー）

《リスト1》『ボール・キャプチャ・ゲーム』のプログラム・リスト（VB6.0用）

```
Option Explicit

Dim keystat%(9) 'テンキーの状態を記録する変数
Dim keytrig0%, keytrig1%, keytrig2%, keytrig3% 'トリガー用キーの状態を記録する変数
Dim playerx%, playery% 'プレイヤーキャラクタの座標
Dim ballx%, bally% 'ボールキャラクタの座標
Dim hiscore%, score% 'ハイスコアとスコア
Dim ti% '残り時間
Dim gamestat% 'ゲーム進行状況

Private Sub Form_Load()
    hiscore = 0
    gameinit
End Sub

Private Sub gameinit() 'ゲーム初期化
    Dim ox%, oy%, x%, y%, i%
    With Form1
        .Width = Screen.TwipsPerPixelX * 640 'フォームのサイズを設定
        .Height = Screen.TwipsPerPixelY * 480
        .Caption = "VBGame02"
    End With
    With Picture1
         .Width = Screen.TwipsPerPixelX * 576 'ピクチャボックスのサイズを設定
        .Left = Screen.TwipsPerPixelY * 32
        .Height = Screen.TwipsPerPixelY * 432
        .Top = 0
        .ScaleMode = 3
        .ScaleHeight = 432
        .ScaleWidth = 576
        .BackColor = RGB(0, 0, 0)
        .ForeColor = RGB(255, 255, 255)
        .FillStyle = 0
        .Font = "ＭＳ ゴシック"
        .FontSize = 8
    End With
    Picture1.Cls
    Form1.Show

    With Picture1
        .CurrentX = 200
        .CurrentY = 0
        .FontSize = 10
        .ForeColor = RGB(255, 255, 255)
    End With
    Picture1.Print "High Score " & hiscore

    playerx = 576 / 2 'プレイヤーキャラクタの初期座標
    playery = 416 / 2 + 16

    Rnd (-1) '乱数系列の初期化
    Randomize 0

    ballx = Rnd * (576 - 20) + 10 'ボールキャラクタの初期座標
    bally = Rnd * (416 - 20) + 10 + 16

    score = 0
    ti = 1000
    putscore

    gamestat = 0 'ゲームモード（0でゲーム中、1でゲームオーバー状態）

    Timer1.Interval = 20 'インターバルタイマ起動
End Sub

Private Sub Picture1_MouseUp(button%, shift%, x!, y!)
    If (button And 2) = 2 Then 'マウス右ボタンクリックで強制終了
        End
    End If
End Sub

Private Sub Picture1_KeyDown(key%, shift%)
    If key = 27 Then 'ESCキーで強制終了
        End
    End If
    If key > 96 And key < 106 Then 'テンキー設定
        keystat(key - 96) = 1
    End If
    If key = 32 Then keytrig0 = 1 'トリガキー設定
    If key = 90 Then keytrig1 = 1
    If key = 88 Then keytrig2 = 1
    If key = 13 Then keytrig3 = 1
End Sub

Private Sub Picture1_KeyUp(key%, shift%)
    If key > 96 And key < 106 Then
       keystat(key - 96) = 0
    End If
    If key = 32 Then keytrig0 = 0
    If key = 90 Then keytrig1 = 0
    If key = 88 Then keytrig2 = 0
    If key = 13 Then keytrig3 = 0
End Sub

Private Sub Timer1_Timer()
    Dim newplayerx%, newplayery%
    If gamestat = 0 Then '通常ゲームモード
        newplayerx = playerx
        newplayery = playery
        If keystat(2) = 1 Then 'キー状況に応じて新規座標を変更
            newplayery = playery + 10
        End If
        If keystat(8) = 1 Then
            newplayery = playery - 10
        End If
        If keystat(4) = 1 Then
            newplayerx = playerx - 10
        End If
        If keystat(6) = 1 Then
            newplayerx = playerx + 10
        End If
        If newplayerx < 10 Then newplayerx = 10 '移動範囲制限
        If newplayerx > 566 Then newplayerx = 566
        If newplayery < 26 Then newplayery = 26
        If newplayery > 422 Then newplayery = 422
         Picture1.Circle (playerx, playery), 10, RGB(0, 0, 0) '元の場所のキャラクタを消去
           Picture1.Circle (newplayerx, newplayery), 10, RGB(255, 255, 255) '新しい位置にキャラクタを描画
        playerx = newplayerx 'プレイヤーキャラクタの座標を更新
        playery = newplayery

        ti = ti - 1 '時間経過
        puttime
        If ti < 1 Then gamestat = 1 '時間切れでゲームオーバー

         Picture1.Circle (ballx, bally), 10, RGB(255, 0, 0) 'ボールを表示

         If distance(playerx, playery, ballx, bally) < 10 + 10 Then 'プレイヤーキャラクタがボールキャラクタと接触したか？
            score = score + 1 '接触したら得点アップ
            putscore
             Picture1.Circle (ballx, bally), 10, RGB(0, 0, 0) 'ボールキャラクタを消去
             ballx = Rnd(1) * (576 - 20) '新しい位置にボールキャラクタを移動させる
            bally = Rnd(1) * (406 - 20) + 16
        End If

    ElseIf gamestat = 1 Then
        With Picture1
            .CurrentX = 230
            .CurrentY = 200
            .FontSize = 15
            .ForeColor = RGB(Rnd * 255, Rnd * 255, Rnd * 255)
        End With
        Picture1.Print "Game Over"
        If score > hiscore Then hiscore = score
        If keytrig3 = 1 Then
            gameinit
        End If
    End If

End Sub

Private Sub putscore()
    Picture1.Line (370, 0)-(469, 15), 0, BF
    With Picture1
        .CurrentX = 370
        .CurrentY = 0
        .FontSize = 10
        .ForeColor = RGB(255, 255, 255)
    End With
    Picture1.Print "Score " & score
End Sub

Private Sub puttime()
    Picture1.Line (470, 0)-(575, 15), 0, BF
    With Picture1
        .CurrentX = 470
        .CurrentY = 0
        .FontSize = 10
        .ForeColor = RGB(255, 255, 255)
    End With
    Picture1.Print "Time " & ti
End Sub

'
'   2点間の距離を求める関数
'
Private Function distance(x0, y0, x1, y1)
    distance = Sqr((x0 - x1) * (x0 - x1) + (y0 - y1) * (y0 - y1))
End Function
```

リスト1 入力&実行方法

①VB起動

まずは，VBを起動します。新しいプロジェクトとして，「標準EXE」を選択して[開く]ボタンをクリックしてください。通常は，最初に標準EXEが選択された状態になっているので，[開く]をクリックするだけです。すると，**写真①**のような画面が表示されます。

②フォームの設定

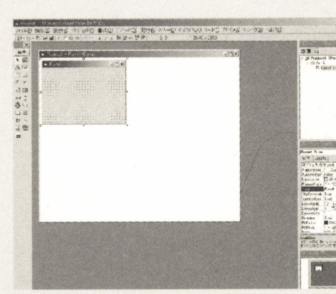
写真②のように，フォームを設定してください。プロパティは，プログラム中で更新されるので設定不要です（不用意に設定すると動作しなくなる場合があるので注意）。設定するのは，PictureBoxとTimerです。それぞれツールボックスのアイコンを選択した後に，フォーム上に貼り付けてください。貼り付ける操作は，フォーム上で左クリックした後，右下方向に少しドラッグしてから左ボタンを離せばOKです。位置，サイズなどは自由です（プログラム中で修正される）。

③リストの入力

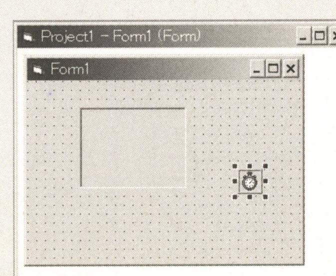
プロジェクト・ウィンドウのコード表示ボタンをクリックすると，コードウィンドウが表示されます。Form1のコードとして，リスト1を入力してください。

④実行

実行メニューから[開始]を選択すると，プログラムが実行されます。



《写真①》VBを起動して「標準EXE」を選ぶ



《写真②》このようにフォームを設定しよう



▲X君

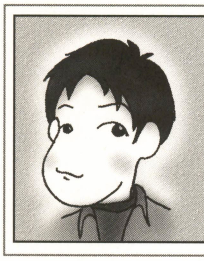
マイクロソフトの技術者がDirectXテクノロジーを伝授する！

IE+DirectAnimation マルチメディア・プログラミング

第2回　テキスト（文字）を描画してみよう！

『Microsoft Internet Explorer（以下IEと略）』と『Direct Animation』を使えば，特別な製品を持っていなくても簡単にマルチメディア・プログラムを作ることができます。そのテクニックを紹介する第2回は，テキスト（文字）を描画する手法でカッコイイ時計を作ってみましょう（写真1）。DirectAnimationは，IEさえあればすぐその場で動作が確認できます。プログラミングに興味のある入門者も，ぜひ挑戦してみてください！

TEXT&PROGRAMING●平井昌人



横浜市生まれの&H25歳男性。1993年4月に，開発言語製品のサポート・エンジニアとしてマイクロソフトに入社。QB，VBを担当し，現在はシステム・エンジニア（SE）としてVisual Studioを中心とした開発系の啓蒙・支援活動を行っている。最近は，『VS.NET（Visual Studioの次期バージョン）』の普及／宣伝活動で，各地を飛びまわっている。先日も，名古屋／大阪へ行ってきた。うーん，関西は食べものがおいしいね。

DirectAnimationとは？

それでは，まず復習からはじめましょう。

DirectAnimation*1は，誰でも知っているテクノロジーとはいえないかもしれません。しかし，単純なプログラムですばらしい視覚的効果を作りだすことができる，優れものなのです。また，大半の人が利用しているIEというブラウザで実行でき，インターネットを通して他の人に見せてあげることもできます。

DirectAnimationは，『Microsoft Visual Basic（以下VBと略）』『Microsoft Visual C++』をはじめ，Office VBAやスクリプト言語で利用することができます。本コーナーでは，『Microsoft Visual Basic Script（以下VBScript*2と略）』を使ったHTMLベースで紹介していきます。

DirectAnimation プログラミングの基礎

それではDirectAnimationについて，プログラミングの基礎を紹介していきましょう。

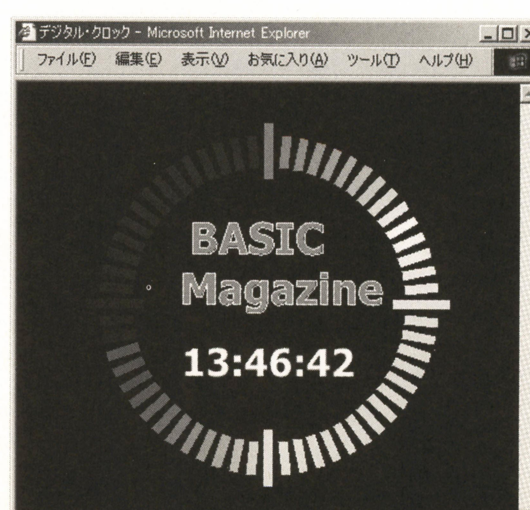
①DirectAnimationコントロール

DirectAnimationを利用する場合は，“Direct Animationコントロール”が必要になります。DirectAnimationコントロールである“danim.dll”“danim.oca”はIEに含まれているので，IEが動いている環境であればすでにインストールされています。

HTMLの中でコントロールを利用するには，以下のように<OBJECT>タグで指定します。コントロールの名前は，“DAControl”にしておきましょう。

```
<OBJECT ID = "DAControl"
   CLASSID = "CLSID:B6FFC24C-
7E13-11D0-9B47-00C04FC2F51D"
   STYLE = "position:absolute;
left:0; top:0; z-index:-1">
</OBJECT>
```

なお，VBで利用する場合は，写真2のように



《写真1》これが今回作成するサンプル『デジタル・クロック』だ。カラーでお見せできないのが残念

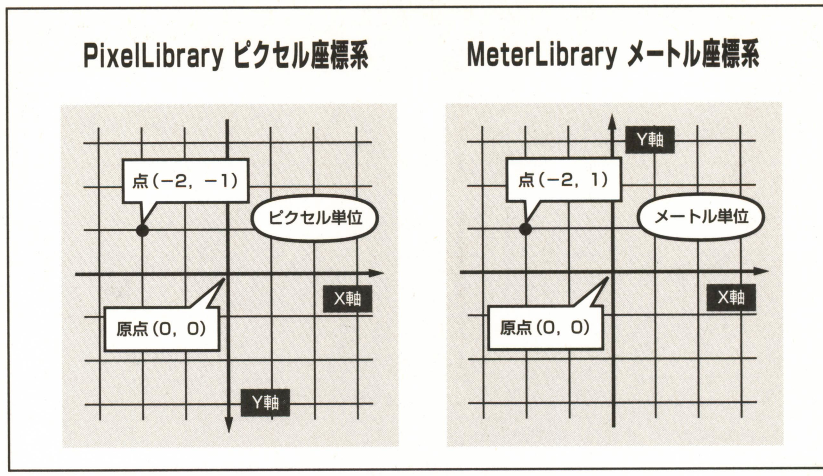
*1　DirectAnimation
2Dや3Dグラフィックス，サウンド，映像などを総合的にサポートする高レベルなAPIを提供している。IEに統合されており，インターネットでの利用を強く意識した設計となっている。

*2　VBScript
IE3.0以降で動作するプログラミング言語で，VBと下位互換性がある。VBScriptのプログラムは，HTMLコード内に埋め込んで使用する。IEでそのHTMLをアクセスすると，組み込まれたVBScriptのプログラムが自動的に実行されることになる。

*3　ピクセル座標系
これを選ぶと，座標の単位がピクセル（Pixel）になる。X軸は右方向が＋，Y軸は下方向が＋となっている。DAControlのPixelLibraryプロパティで初期化する。ピクセル座標系は，Windowsと同じ単位なので，2Dグラフィックスを利用する場合に便利。



《写真2》DirectAnimationをVBで利用する場合は，ここでコントロールの追加を行っておこう



《第1図》DirectAnimationの座標系は二つあるので注意

VBの[プロジェクト]メニューの[コンポーネント]で“DirectAnimation Library”にチェックを付けます。つぎに，[ツールボックス]から“DA ViewerControl”コントロールをフォームに貼り付けて，[プロパティ]ウィンドウのオブジェクト名をDAControlに変更します。プログラムは，VBScriptとほとんど同じです。

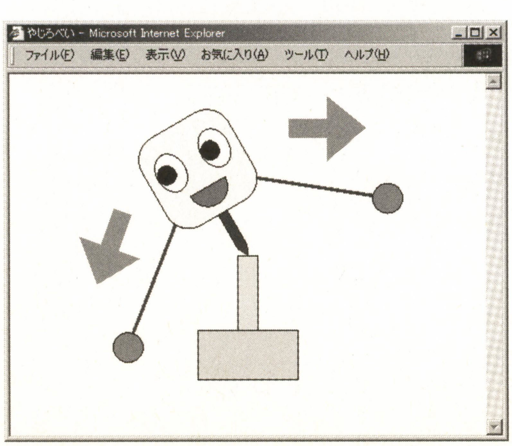
②VBScriptのコードを記述

HTML内にVBScriptのコードを記述する場合は，つぎのように<SCRIPT >タグから</SCRIPT>のあいだに記述します。

```
<HTML>
<BODY LEFTMARGIN = "0", TOPMARGIN =
"0">
<SCRIPT LANGUAGE = "VBScript">
  '// ここにプログラムコードを記述する
</SCRIPT>
```

③DirectAnimationの初期化

はじめにDirectAnimationを初期化して，座標系ライブラリを選択します。DirectAnimationコントロールの座標は，原点(0, 0)がつねにコントロールの中心にきます。そして，**第1図**のように“ピクセル座標系[*3]”と“メートル座標系[*4]”の二つが用意されており，初期化のときにどちらを使うのか選択しなければなりません。



《写真3》2001年4月号で紹介したサンプル『やじろべえ』

初期化のコードは，つぎのようになります。変数mは，DirectAnimationのプロパティやメソッドを利用するためのDAStaticsオブジェクト[*5]が入ります。

```
Set m = DAControl.PixelLibrary
```

④イメージの作成

DirectAnimationでは，“イメージ(DA Image[*6])”と呼ばれるオブジェクトを作らないと何も表示されません。DirectAnimationは，数々のメソッドを使い多彩なイメージを作りあげ，そのイメージの色を変化させたり動きをあたえたりして加工します。そして，加工した複数のイメージを重ね合わせて，最終的に1枚のイメージを完成させるというスタイルをとります。

つぎのコードは，SolidColorImageというメソッドにm.Black(黒)を渡して，「黒のベタ塗りイメージ」を作成します。

```
Set img = m.SolidColorImage(m.Black)
```

⑤イメージをコントロールにセットしてスタート

変化や動きをあたえて加工した複数のイメージは，合成を繰り返して最終的に一つのイメージにまとめます。最終イメージは，DirectAnimationコントロールのImageプロパティに代入して，Startメソッドを呼びだします。

```
DAControl.Image = img
DAControl.Start
```

これがテキスト(文字)の描画テクニックだ！

先月号では，**写真3**のように単純な図形のイメージをたくさん作り，それを合成するテクニ

＊4　メートル座標系
これを選ぶと，座標の単位がメートル（m）になる。X軸は右方向が＋，Y軸は上方向が＋となっている。DAControlのMeterLibraryプロパティで初期化する。メートル座標系は，主に3Dを取り扱う場合に利用されるのだ。

＊5　DAStaticsオブジェクト
DirectAnimationの一番上位に位置するオブジェクト。このオブジェクトが提供するプロパティやメソッドを利用して，また別のオブジェクトを生成していく。このオブジェクトは利用回数が多いので，変数名は短くしておこう。MeterLibraryのときは“m”，PixelLibraryのときは“p”とでもしておくといいのかもしれないが，本誌ではmに統一している。

＊6　DAImageオブジェクト
DirectAnimationで，DAStaticsオブジェクトのつぎに利用頻度が高い，重要なオブジェクト。DAImageオブジェクトは，メモリ上に1枚の2Dグラフィックス・イメージを保持／管理している。DirectAnimationでは，このDAImageオブジェクトを作成し，加工や変形を加え，イメージ同士を合成していくという流れのプログラミングとなる。

《第1表》DAFontStyleオブジェクトを返すメソッドとプロパティ

メソッド／プロパティ	引数	内容
m.DefaultFont	なし	デフォルトのDAFontStyleオブジェクト。“Times New Roman”，12ポイント，黒
m.Font(face, point, color)	face	フォント名（String）
	point	サイズ（Double）
	color	色（DAColor）
f.Family(face)	face	フォント名（string）の設定
f.Size(point)	point	サイズ（Double）
f.Color(color)	color	色（DAColor）
f.Bold	なし	太字[ボールド]
f.Italic	なし	斜体[イタリック]

※mはDAStaticsオブジェクト，fはDAFontStyleオブジェクトを示す。

ックを紹介しました。今回は，テキストを描画する方法を解説します。

DirectAnimationでは，図形と同じように，文字も一つのイメージ（DAImageオブジェクト）に仕上げていきます。

DAFontStyleオブジェクト～Fontメソッド

テキストを描画するには，どのようなスタイル（フォント名，サイズ，色など）を使うか，指定する必要があります。

DirectAnimationでは，DAFontStyleオブジェクトで制御します。DAFontStyleオブジェクトは，Fontメソッドを使って作成します。

つぎのコードは，「フォント名がArial，サイズが24ポイント，色が黄色」のDAFontStyleオブジェクト（fnt）を作成します。

```
Set fnt = m.Font("Arial", 24, m.Yellow)
```

もちろん，太字にしたり，斜体にしたりすることもできます。**第1表**に，DAFontStyleオブジェクトを返すメソッドとプロパティの一部を示しておきます。

文字列からイメージを作成する2とおりの方法

さて，「ABC」のような文字列から，「ABC」が描かれたイメージを作成する方法は，大きくわけてつぎの2とおりがあります。

①文字列から直接イメージを作成する方法～StringImageメソッド

文字列を，イメージ（DAImageオブジェクト）に直接変換します。この方法が一番簡単です。StringImageメソッドに「ABC」などの文字列とDAFontStyleオブジェクトを渡し，描画されたDAImageオブジェクトを受け取ります。

文字列からイメージを作成する①

```
Set img = m.StringImage("Hello World", fnt)
```

Hello World

《第2図》StringImageメソッドを使って，このように文字列から直接イメージを作成する

文字列からイメージを作成する②

```
Set txt = m.StringPath("Hello World", fnt)
Set img = txt.Draw(border)
```

Hello World

《第3図》StringPathメソッドを使って，文字列を軌跡に変換してから描画する（Drawメソッドを使う場合）

文字列からイメージを作成する③

```
Set txt = m.StringPath("Hello World", fnt)
Set img = txt.Fill(border, paint)
```

Hello World

《第4図》StringPathメソッドを使って，文字列を軌跡に変換してから描画する（Fillメソッドを使う場合）

つぎのコードは，**第2図**のようになります。

```
Set img = m.StringImage("Hello World", fnt)
```

②文字列を軌跡に変換してから描画する方法～StringPathメソッド

この方法は，文字列を一度，軌跡（DAPath2オブジェクト*7）へ変換してからイメージ（DAImage）を作成します。これは，先月4月号で紹介した図形の描画と同じ考えかたです。あらかじめ，枠線のスタイルを決定するDALineStyleオブジェクトや，塗りつぶしのイメージを決定するDAImageオブジェクトを用意しておく必要があります。

つぎのコードは，文字列「Hello World」の軌跡を作成します。しかし，まだこの段階ではイメージになっていません。変数txtは，DAPath2オブジェクトです。

```
Set txt = m.StringPath("Hello World", fnt)
```

つぎに，軌跡からイメージを作成します。DrawメソッドかFillメソッドの，どちらかで描画します。

●Drawメソッド（枠のみ描画）

Drawメソッドを使って，**第3図**のように青の枠だけで描画します。

＊7 DAPath2オブジェクト

図形や線，文字などの“軌跡”を管理するオブジェクト。軌跡とは，数学でいう一次変換で表現できる点や線の集合体。点や線を表す方程式のようなものを管理している。簡単にいえば，DAPath2オブジェクトは図形や文字を描画するための“型紙”である。軌跡（DAPath2オブジェクト）からイメージ（DAImageオブジェクト）を描画するしくみは，型紙にそって鉛筆でなぞるとその図形が書けるのと同じ。

特別記念品付き DirectAnimationプログラム投稿受付中！

みなさんからのDirectAnimation／DirectX for Visual Basicプログラムの投稿をお待ちしております。投稿されたプログラムは，マイクロソフト平井さんに審査していただき，優秀作品は特選投稿プログラムコーナーに掲載されます。なんと，規定原稿料のほかに，マイクロソフト社から提供される『Visual Studio 97オリジナルTシャツ』をプレゼントします！　どしどしご応募ください。

■投稿要項

DirectAnimationを使ったプログラムは，できるだけVBScriptで書かれたものが対象です（どうしてもできない場合はVBも可）。DirectX for Visual Basicは，VBのソースおよびexeファイルを同送してください。画像，サウンドなどは，著作権および肖像権などを侵害しないように注意しましょう。

投稿のきまりは，特選プログラムコーナーに準じますが，電子メールでも受け付けます。

■あて先

〒141-8715東京都品川区東五反田1-11-15電波新聞社マイコンBASICマガジン「特選プログラムコーナーDirectAnimation」係
電子メール：webmas@basicmagazine.net

```
Set border = m.DefaultLineStyle.
Color(m.Blue)
Set img = txt.Draw(border)
```

●Fillメソッド(枠と内部を描画)

Fillメソッドを使って，第4図のような青の枠と内部を黄色の塗りつぶしで描画します。

```
Set border = m.DefaultLineStyle.
Color(m.Blue)
Set paint = m.SolidColorImage
(m.Yellow)
Set img = txt.Fill(border, paint)
```

サンプル『デジタル・クロック』を実行してみよう！

DirectAnimationで，テキスト(文字)の描画をテーマにしたサンプル『デジタル・クロック*8』“digital.htm”を用意しました。ソースコードは，リスト1です。なお，サンプルのソースコードは，ベーマガの公式ホームページにありますのでダウンロードしてください。

●http://www.basicmagazine.net/

サンプルを実行すると，写真4のような幻想的な時計のアニメーションが表示されます。カラーでお見せできないのが残念ですが，とってもきれいですよ。

時計の中央に,「BASIC Magazine」と表示されている部分と，HH:MM:SS形式で時刻が表示されている部分が今回のテーマとなっています。



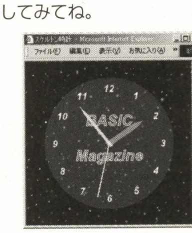
《写真4》サンプル・プログラム『デジタル・クロック』の動き

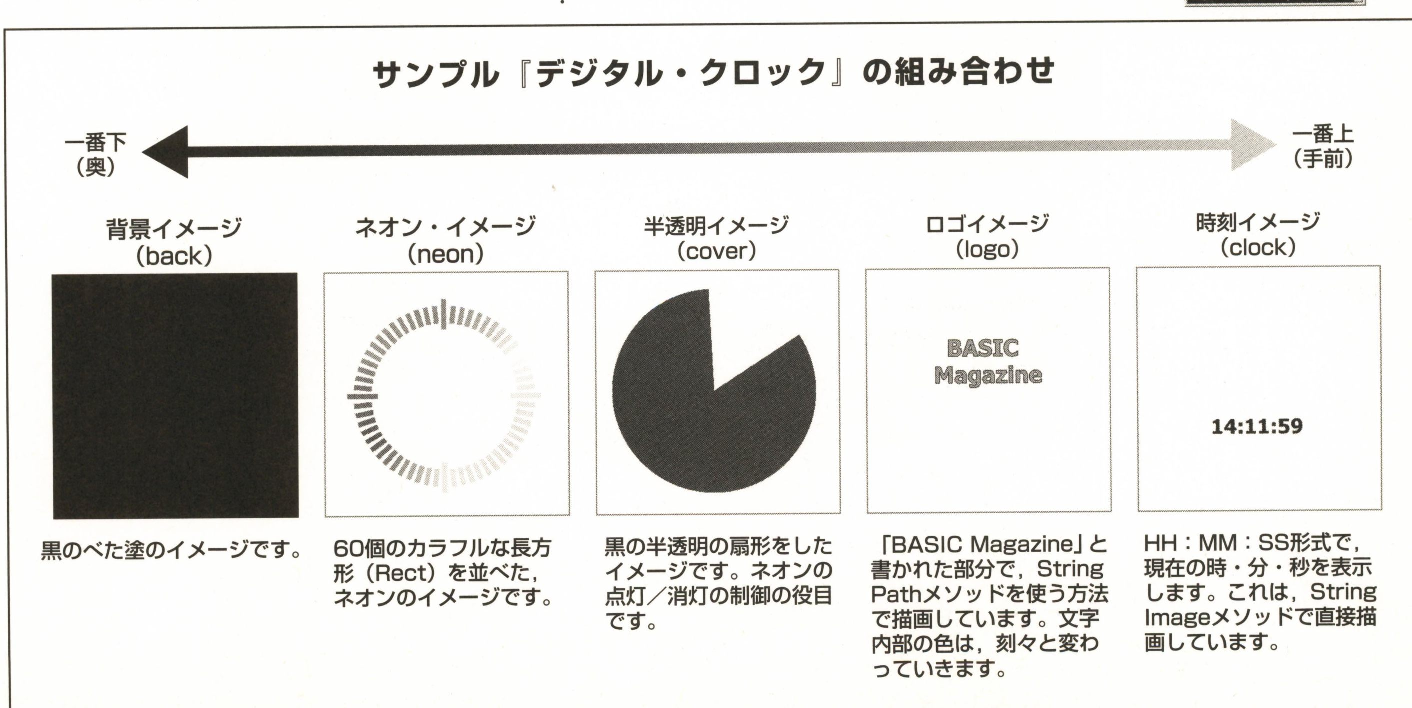
サンプルの仕組みを見てみよう！

このサンプルは，第5図のような五つのパーツ(イメージ)を組み合わせています。プログラムで描画する一番下(奥)から，表の順で合成されています。

サンプルをよく見ると，TV番組の『タイムショック』のように，秒と連動して外側のネオンが点灯していきます。60秒経つと全部消えて，

＊8　デジタル・クロック
このサンプルは，横浜の『みなとみらい21』にある大きな観覧車を思いだして作ってみたもの。1998年6月号では，下の写真のようなちゃんとした時計も作ってある。サンプルがホームページにまだあると思うから試してみてね。





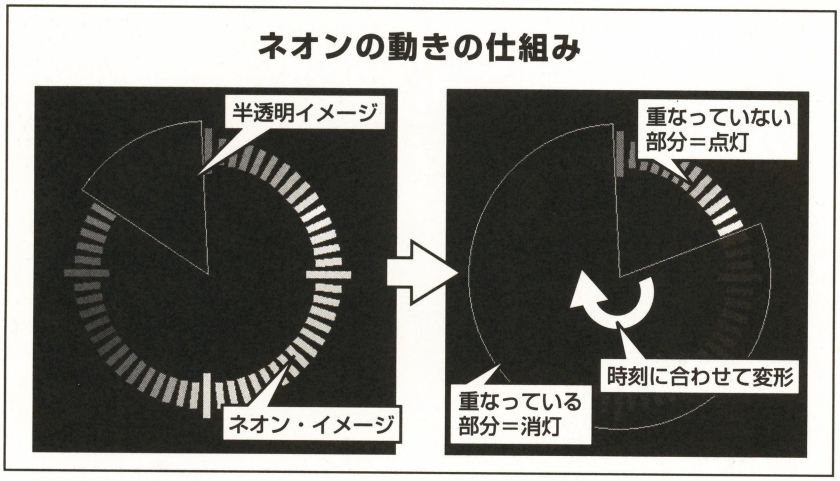
《第5図》デジタル・クロックを構成するパーツ(イメージ)は，5階層になっている

またはじめにもどります。

このアニメーションは，ネオンの色を変えているのではなく，第6図のようにあらかじめ全部点灯した状態のイメージのうえに，黒の半透明の扇形イメージをかぶせています。このため，かぶさっているところは消灯，かぶさっていないところは点灯しているように見えるのです。そして，半透明の扇形の終了角を秒に合わせて変化させているので，順番にネオンが点灯していくように見えます。

また，ここでは半透明（スケルトン）の設定を行っています。DAImageオブジェクトの透明度は，つぎのコードのようにOpacityメソッド*9で設定します。引数は0.0～1.0の範囲で，0.0が完全透明，1.0が不透明，そのあいだの0.5などの数値で半透明（スケルトン）となります。

```
Set img = img.Opacity(0.5)
```



《第6図》ネオンのオン／オフは，このような仕組みになっている

まとめ

DirectAnimationは，少ないコードでクオリティの高いアニメーションを簡単に実現することができます。また，プログラミングやアイデアしだいで，今回のサンプルのように凝ったものも作れます。ぜひとも，みなさんも挑戦してみてください。

また，投稿プログラミングも，楽しみにお待ちしております！

リスト1入力&実行方法

このリストは，IE5／5.5とテキスト・エディタ（『メモ帳』など）があれば，実行することが可能だ。IE5は，以下のアドレスから無償でダウンロードすることができる。

●http://www.microsoft.com/windows/ie_intl/ja/download/

リストをメモ帳などで入力したら，拡張子を“htm”または“html”にして保存する。そして，IE5の[ファイル]メニューより，[開く]で保存したファイルを読み込もう。すると，リストが実行されるようすを，IE5で確認することができる。

＊9 Opacityメソッド
イメージ（DAImage）の透明度を設定するメソッド。スケルトン効果やフェード・イン／フェード・アウトが演出できる，便利なメソッドだ。内部では，アルファ・ブレンディング処理が行われている。Windows 2000やWindows Meなどで，メニューが「ボワァ」と浮かんで表示される機能があるが，あれも同じ原理。なんと，次期バージョンのVBは，フォームのプロパティとして装備している。

《リスト1》サンプル『デジタル・クロック』“digital.htm”のプログラム・リスト（VBScript用）

```
<HTML>
<TITLE>デジタル・クロック</TITLE>
<BODY LEFTMARGIN = "0", TOPMARGIN = "0">
<SCRIPT LANGUAGE = "VBScript">
'//-----------------------------------------------------------
'//
'// DirectAnimation Sample                         by Masato Hirai
'//
'//    『デジタル・クロック』
'//
'//-----------------------------------------------------------
Option Explicit

Sub Window_onLoad()
  '//
  '// 変数・定数の宣言
  '//
  Const PI = 3.14159265359          '円周率
  Const FONT_NAME = "Tahoma"        'フォント名
  Dim m        'As DAStatics        'DAStaticsオブジェクト
  Dim tm       'As DANumber         '時刻
  Dim h1       'As DAString         '時(10の位)
  Dim h2       'As DAString         '時(1の位)
  Dim m1       'As DAString         '分(10の位)
  Dim m2       'As DAString         '分(1の位)
  Dim s1       'As DAString         '秒(10の位)
  Dim s2       'As DAString         '秒(1の位)
  Dim border   'As DALineStyle      '線のスタイル
  Dim paint    'As DAImage          '塗りつぶしイメージ
  Dim fnt      'As DAFontStyle      'フォント
  Dim HMS      'As DAString         'HH:MM:SS
  Dim clock    'As DAImage          '時刻イメージ
  Dim logo     'As DAImage          'ロゴイメージ
  Dim cover    'As DAImage          '半透明イメージ
  Dim neon     'As DAImage          'ネオンイメージ
  Dim back     'As DAImage          '背景イメージ
  Dim col      'As DAColor          'DAColorオブジェクト
  Dim num      'As DANumber         'DANumberオブジェクト
  Dim img      'As DAImage          'DAImageオブジェクト
  Dim n        'As Long
  '//
  '// DirectAnimationの初期化 (ピクセル座標系)
  '//
  Set m = DAControl.PixelLibrary
  '//
  '// HH:MM:SSの動的な文字(DAStringオブジェクト)の作成
  '//
  Set tm = m.Add(m.DANumber(Timer), m.LocalTime)
  '// 時(HH)
  Set num = m.Mod(m.Div(tm, m.DANumber(3600)), m.DANumber(24))
  Set h1 = m.Floor(m.Div(num, m.DANumber(10))).ToString(0)
  Set h2 = m.Floor(m.Mod(num, m.DANumber(10))).ToString(0)
  '// 分(MM)
  Set num = m.Mod(m.Div(tm, m.DANumber(60)), m.DANumber(60))
  Set m1 = m.Floor(m.Div(num, m.DANumber(10))).ToString(0)
  Set m2 = m.Floor(m.Mod(num, m.DANumber(10))).ToString(0)
  '// 秒(SS)
  Set num = m.Mod(tm, m.DANumber(60))
  Set s1 = m.Floor(m.Div(num, m.DANumber(10))).ToString(0)
  Set s2 = m.Floor(m.Mod(num, m.DANumber(10))).ToString(0)
  '//
  '// 時刻イメージ
  '//
  Set border = m.DefaultLineStyle.Color(m.Yellow)
  Set fnt = m.Font(FONT_NAME, 34, m.White).Bold
  Set HMS = m.ConcatString(m.ConcatString(h1, h2), _
            m.ConcatString(m.ConcatString(m.DAString(":"), _
            m.ConcatString(m1, m2)), _
            m.ConcatString(m.DAString(":"), _
            m.ConcatString(s1, s2))))
  Set clock = m.StringImageAnim(HMS, fnt)
  Set clock = clock.Transform(m.Translate2(0, 60))
  '//
  '// 飾り文字イメージ
  '//
  Set num = m.Mul(m.LocalTime, m.DANumber(0.1))
  Set col = m.ColorHslAnim(num, m.DANumber(1), m.DANumber(0.65))
  Set paint = m.SolidColorImage(col)
  Set border = m.DefaultLineStyle.Color(m.White)
  Set fnt = m.Font(FONT_NAME, 38, m.White).Bold
  '// "BASIC"
  Set img = m.StringPath("BASIC", fnt).Fill(border, paint)
  Set num = m.Mul(m.Sin(m.LocalTime), m.DANumber(16))
  Set img = img.Transform(m.Translate2Anim(num, m.DANumber(0)))
  Set logo = img.Transform(m.Translate2(0, -42))
  '// "Magazine"
  Set img = m.StringPath("Magazine", fnt).Fill(border, paint)
  Set num = m.Mul(m.Sin(m.LocalTime), m.DANumber(-16))
  Set img = img.Transform(m.Translate2Anim(num, m.DANumber(0)))
  Set logo = m.Overlay(img, logo)
  '//
  '// ネオンイメージ
  '//
  Set neon = m.EmptyImage
  Set border = m.EmptyLineStyle
  For n = 0 To 59
    Set paint = m.SolidColorImage(m.ColorHsl(n / 60, 1, 0.6))
    If (n Mod 15) = 0 Then
      Set img = m.Rect(48, 8).Fill(border, paint)
    Else
      Set img = m.Rect(28, 8).Fill(border, paint)
    End If
    Set img = img.Transform(m.Translate2(128, 0))
    Set img = img.Transform(m.Rotate2Degrees(n * -6 + 90))
    Set neon = m.Overlay(img, neon)
  Next
  '//
  '// 半透明イメージ
  '//
  Set paint = m.SolidColorImage(m.Black).Opacity(0.8)
  Set num = m.Sub(m.DANumber(PI * 2), _
            m.Mul(m.DANumber(PI / 30), _
            m.Floor( _
            m.Mod(tm, m.DANumber(60)))))
  Set border = m.EmptyLineStyle
  '// 終了角に変動値を用いた扇形
  Set cover = m.PieRadiansAnim( _
                m.DANumber(PI / 30), _
                num, _
                m.DANumber(320), _
                m.DANumber(320)).Fill(border, paint)
  Set cover = cover.Transform(m.Rotate2Degrees(90 - 3))
  '//
  '// 背景イメージ
  '//
  Set back = m.SolidColorImage(m.Black)
  '//
  '// イメージの合成
  '//
  Set img = m.OverlayArray(Array(clock, logo, cover, neon, back))
  '//
  '// コントロールにオブジェクトをセットしてスタート
  '//
  DAControl.Image = img
  DAControl.Start
  Call Window_onResize()
End Sub

'-----------------------------------------------------------
'ウィンドウのサイズ変更時の処理
'-----------------------------------------------------------
Sub Window_onResize()
  On Error Resume Next
  DAControl.Width  = Document.Body.ClientWidth
  DAControl.Height = Document.Body.ClientHeight
End Sub

</SCRIPT>
<! DirectAnimation コントロール>
<OBJECT ID = "DAControl"
  CLASSID = "CLSID:B6FFC24C-7E13-11D0-9B47-00C04FC2F51D"
  STYLE = "position:absolute; left:0; top:0; z-index:-1">
</OBJECT>
</BODY>
</HTML>
```

PlayStation 2でゲームが作れる！



新連載

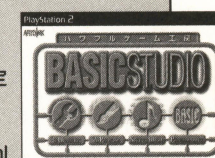
「BASIC STUDIO パワフルゲーム工房」入門

第1回 簡単なBASICコマンドを使ってみる

文／山崎　寛

本誌3月号，4月号と連続して紹介したPlayStation 2用ソフト『BASIC STUDIO』ですが，残念なことに発売が2001年4月下旬に延期となってしまいました。楽しみにしている人も多いと思いますが，もうしばらく待ちましょう。編集部も期待してるんです。

『BASIC STUDIO パワフルゲーム工房』
発　　売：アートディンク
対応機種：PlayStation 2
メディア：CD-ROM 2枚組
発 売 日：2001年4月下旬予定
価　　格：9,800円（税抜）
http://www.artdink.co.jp/japanese/title/bs/index.html



プログラムって難しい？

この連載は，PS2でプログラミングを可能にした『BASIC STUDIO パワフルゲーム工房』を使い，実際にゲームを作っていこうというものです。

第1回目の今回は，発売前にまずはBASIC言語ってどんなものかを説明してみたいと思います。

「今までプログラムというのは全然したことなくて，買ったけど何をしたらいいかさっぱり」なんて方もいるかもしれませんが，ここではそうしたプログラムはまったく初めてという人でも楽しめるように，BASICの基礎の基礎を教えたいと思います。できれば実際に読みながら，BASICを使ってみるとわかりやすいでしょう（ソフト購入してからでも）。

コンピュータは命令を待っている

キーボードを接続してBASIC STUDIOを立ち上げたら，EZ-BASICを選んでください。すると，

```
EZ-BASIC Version ○○
OK
```

と書かれた画面になり，「OK」の下に白い四角が表示されていると思います。この四角が「カーソル」と呼ばれるもので，キーボードのキーを押すと常にこのカーソルの場所に文字が表示され，カーソルは右にひとつずつ移動していきます。

「OK」というのは，コンピュータ（この場合はPS2ですね）がユーザー（あなたのことです）からの「命令」を待っている状態のことで，BASICを立ち上げたらプログラムなどの実行中以外は常にこの「OK」という状態になっています。

「命令」というのは，「ああしろ」とか「こうしろ」とかいうことで，つまりコンピュータは人間の指図をじっと待っていて，命令が来た途端その指示を忠実にこなすわけです。

それでは試しにコンピュータへ命令してみましょうか。

```
PRINT 1
```

と入力して，[Enter]キーを押します。

すると，画面に1が表示されてまた「OK」が表示されますね。

今書いたPRINTというのは「画面に○○を表示しなさい」という命令のことで，今は「1」を後ろにつけたので「1」が画面に表示されました。そして，コンピュータは与えられた命令を実行し終えたので，再び「OK」を表示してつぎの命令を待っているというわけです。

続いて簡単な計算をさせてみましょう。

```
PRINT 1+2
```

など，簡単な計算式をPRINTの後ろに書いて[Enter]キーを押してみてください。今度は計算の結果が表示されたと思います。いろいろな計算をさせてみるのも面白いでしょう。そうそう，計算は「＋」と「－」はそのままですが，「×」は「＊」，「÷」は「／」とそれぞれBASICでは表します，気をつけましょう。

このように，PRINT文でBASICを簡単な電卓として使えることがわかったと思います。じつはカッコなども使えるので，普通の電卓より便利に使えることもあります。

数字と文字

今度は計算ではなく，名前などの文字を表示させてみましょう。

```
PRINT DEMPA
```

などと好きな言葉を書いてみてください。どうでしょう，「DEMPA」と表示されたでしょうか？

たぶんちゃんと表示されずに，画面に表示されたのは「0」となってしまったと思います。

くわしくは後に回しますが，これはPRINTの後に書いた「DEMPA」が文字列だとわからないためです。文字列だとわからせるためには，その前後に「”」の記号をつけます（この記号を入力するには，[Shift]キーを押しながら[2]のキーを押します。またこの記号をダブルクォーテーションマークといいます）。

それではもう一度試してみましょう。

```
PRINT "DEMPA"
```

今度は正しく「DEMPA」と表示されましたね。このように，BASICで文字列を扱う場合には前後に「"」をつけるということを覚えておきましょう。

ダイレクト・モードとプログラム

これまでのように，「命令」をそのまま書いて[Enter]を押し，すぐに命令の結果をだすことを「ダイレクト・モード」などといいます。

ダイレクト・モードでは，電卓代わりに手軽にコンピュータを使うことができますが，電源を切ったり，入力した命令が画面から消えてしまうとその内容は失われてしまいますし，たくさんの命令を順番に動かすといった複雑なことはできません。

そこで，今度はプログラムにしてみましょう。プログラムというのは，命令を実行する手順を記述したものです（**第1図**）。コンピュータは，プログラムを読みながら手順のとおりに一つずつ命令を実行します。

プログラムにするには，命令の前に行番号と呼ばれる数字をつけるだけです。試してみましょう。

```
10 PRINT 2*3
```

頭の10が行番号です。[Enter]を押してみると，今までなら結果とOKが表示されたのに，何も表示されずにカーソルがつぎの行の頭に来るだけですね。

これで，今入力した命令がプログラムとしてメモリ内に記憶されました。この内容を見るには

```
LIST
```

と入力して[Enter]キーを押してください。画面から消えていても，いつでも入力した内容が現れるはずです。

また，このプログラムを実行するには

```
RUN
```

と入力して[Enter]キーを押します。さっきのプログラムでは「6」と表示されたことでしょう。

今のプログラムは1行だけなのでダイレクト・モードとあまり変わった感じがしませんが，今度はもう2行追加して3行にしてみましょう。

```
20 PRINT "3+5*6="
30 PRINT 3+5*6
```

と入力して，LISTで内容を確認してみましょう。10行から30行までが表示されますね。このように行番号をつけて入力していくことにより，どんどんプログラムを追加していくことができます。また，常に行番号の順番に並ぶので，たとえば10行と20行の間にプログラムを追加したければ，15行を後から追加することで可能です（行番号を10きざみで使うのは，このように後から追加できるようにするためです）。

また，メモリに記憶されているプログラムを修正するには，LISTでプログラムを表示してからカーソル・キー（矢印の書いてあるキーです）でカーソルを直したい場所に移動，最後に[Enter]キーを押すことで修正できますし，行全体を新しくしたい場合には同じ行番号のプログラムを入力し直せば前の内容と入れ替わります。

プログラムすべてを消して，一からやり直す場合，

```
NEW
```

というコマンドですべてのプログラムが消えます。消してしまうともう戻せないので，保存する前のプログラムなどは間違って消さないように注意が必要です。

変数は箱

つぎに「変数」という，プログラムでとても重要なことについて説明します。

変数というのは，「箱」のようなものです（**第2図**）。この箱はコンピュータの中にたくさん用意することができて，それぞれの箱には好きな名前をつけることができます。そして，箱には数字を入れる箱や文字を入れる箱といった種類があります。

変数の中に数字や文字を入れるには，「変数 = 内容」という形で記述します。

たとえば，

```
A=1
```

とすると，変数Aの内容が1になります。

続けて，

```
B=A+2
```

とした場合，Aには1が入っていますから，1＋2で変数Bには3が入ることになります。確かめてみましょう。

```
PRINT B
```

これで変数Bの内容が表示されます。「3」と表示されましたね？

つぎに文字を変数に入れてみましょう。文字を入れるための変数には，変数名の後ろに「＄」記号をつけて数値を扱う変数とは区別します。

```
A$="BASIC-MAGAZINE"
```

などとすれば，変数A$に「BASIC-MAGAZINE」という内容が入ります。

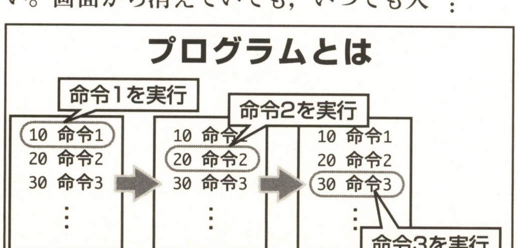
```
PRINT A$
```

で確かめてみればそのとおりに表示されるはずです。試してみてください。

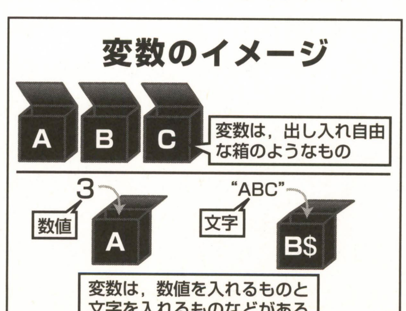
INPUTで入力

今度はプログラム実行中に入力するようにしてみましょう。

INPUT文を使うと，そこでキーボードから入力があるまで一旦プログラムの動作がストップし，入力された内容を変



《第1図》プログラムは命令を順番に並べたもので，一つずつ順番に実行されていく



《第2図》数字や文字を一時記憶させる場所として変数が存在する

数に入れて続きを実行します。
たとえば,

```
INPUT A
```

とした場合，変数Aに入れる数値をBASICが聞いてきますので，キーボードから数値を入力して[Enter]キーを押します。その後,

```
PRINT A
```

で確かめてみれば，確かに入力した数値が変数Aに入っていることを確認できるでしょう。

INPUTを使うと，同じような計算をするプログラムでも，**リスト1**のように対話型のプログラムを作ることができます。これによって，数値を変えるためにいちいちプログラムを直すことなく，同じプログラムで違う計算ができるわけです。

条件によって判断させる

コンピュータが条件になっているかを自分で判断して，それによって違う処理をするための命令がIF〜THENです。
この命令は,

```
IF 条件式 THEN 命令
```

という形で使い，条件式に書いた条件を満たす場合はTHENの後ろの命令を実行し，そうでない場合は実行しません。

条件式というのは，変数でも式でもいいのですが，通常**第1表**のように等号や不等号を使って条件を記述します。

IF文のたとえですが，つぎのように記述した場合，変数Aが10より小さい場合に変数Aの内容を表示します。

```
IF A<10 THEN PRINT A
```

つぎに，INPUT文と組み合わせてプログラムを作ってみましょう。**リスト2**は，INPUT文によって入力された数値が0より大きいか，それとも小さいかを判断します。

《リスト1》計算する数値をINPTUで聞いてくるプログラム・リスト

```
10 INPUT A
20 INPUT B
30 PRINT A+B
```

《第1表》条件式の内容

条件式	条件が満たされる場合
A ＝ B	AとBが等しい
A<>B	AとBは等しくない
A ＜ B	BよりAのほうが小さい
A<=B	AはB以下である
A ＞ B	BよりAのほうが大きい
A>=B	AはB以上である

《リスト2》IF文で数値を判断するプログラム・リスト

```
10 INPUT A
20 IF A>0 THEN PRINT "PLUS"
30 IF A<0 THEN PRINT "MINUS"
```

決まった数の繰り返し

突然ですが,「1から10まで足した数を表示するプログラム」を作りなさいと言われた場合，どうしますか？

```
10 PRINT 1+2+3+4+5+6+7+8+9+10
```

でもいいですし，数学の公式を利用して

```
10 PRINT (10*10+10)/2
```

とも記述できます。実際のところ公式を利用するのが一番スマートな書きかただったりしますが，ここではプログラム的な書きかたを紹介することにしましょう。

FOR文という命令があるのですが，この命令を使うと簡単に一定回数の繰り返しを実現することが可能です。

FOR文を使うには必ずループの回数を示す変数を使います。この変数をAとした場合，FOR文の使いかたは

```
FOR A=最初の値 TO ループ終了の値
ループの処理
NEXT A
```

という形になります。

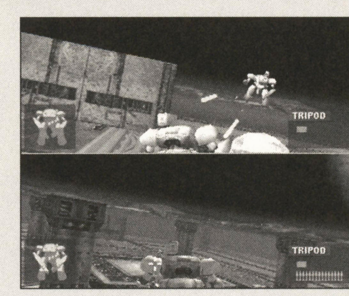
これを実行すると，最初にFORが実行されたときに，変数Aには最初の値が代入されます。

そしてつぎの行にあるループの処理が実行され，続いてNEXT文を実行します。

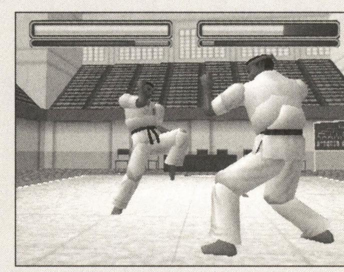
NEXT文では，変数Aがループ終了の値になっているかを判断して，なっていない場合変数Aを＋1してFOR文のつぎの命令に戻ります。

こうして，変数Aが条件を満たす値になるまでの回数ループの処理を繰り返すわけです。

「BASIC STUDIO」でこんなゲームが作れる!!



▲BASIC STUDIOでは，画面を縦や横に2分割して，それぞれを別のカメラから見た3D視点表示することが可能だ。そのため，このような2人対戦型の3Dゲームや，同様に画面分割したレースゲームなども作成可能だろう



▲3Dの対戦格闘も充分遊べるスピードで動作する。こういった人体の動きを作っていくのは容易ではないが，がんばればBASIC STUDIOでも作成可能だ。また，これらサンプルは製品に含まれている

このため，1から10まで足すには

```
10 FOR A=1 TO 10
20   B=B+A
30 NEXT A
40 PRINT B
```

と記述すればいいわけです。実際に試してみましょう。ちゃんと結果が表示されましたね。

ではまた来月

今月は，BASICの基本コマンドを紹介してみました。早くゲームが作りたいよと思う人もいるかもしれませんが，あわてない，あわてない。「BASICでプログラムする」とはどんな感じなのか，まずは理解してもらえたでしょうか。

「BASIC STUDIOパワフルゲーム工房」には，とても親切なBASIC入門マンガ（100ページを超えている！）がついていますので，ぜひ読んでみてください。



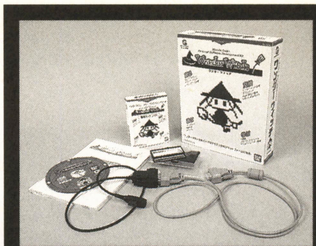
ワンダースワンのソフトを自作しよう!

チャレンジ!! ワンダーウィッチ

ワンダースワン ホビープログラミングツール WonderWitch ワンダーウィッチ

Challenge WonderWitch

第7回「シナリオのスクリプト化」



ワンダースワン・ホビープログラミングツール
『ワンダーウィッチ（WonderWitch）』
定価16,800円（税別）

◆パッケージ内容
・PC-WonderSwan 接続ケーブル
（スワンケーブル，クロスケーブル）
・ワンダーウィッチ専用カートリッジ
・CD-ROM　・マニュアル

◆動作環境
機種: PC/AT 互換機
OS: Windows 98／Me／NT4.0／2000
CPU: Pentium 133MHz以上
メモリ: 32MB以上（推奨64MB以上）
ディスプレイ: 640×480，256色以上
シリアルポート: 9600/38400bps
＊専用カートリッジのみ単体の購入も可能。3,980円（税別）。

◆製品について問い合わせ
wonderwitch-info@bandai.co.jp

◆通信販売について問い合わせ
netshop@bandai.co.jp

今月はワンダーウィッチのプログラムコンテスト「WWGP2001」の結果発表などを紹介します。C言語の勉強にワンダーウィッチ!!（松原拓也）

■オリジナルスクリプトを考案

先月号で作ったアドベンチャー・ゲーム「キュートちゃんの冒険（2001年4月号参照）」はプログラム的に少々問題がありました。ゲーム中の場面ごとに“switch～case”や“if”文が羅列されていて，無駄があったからです。そこで今回は，シナリオのスクリプト化に挑戦してみましょう。つまり，ゲームプログラム自身がシナリオを解読して選択肢や画面表示を行えるようにすればいいのです。

ではまず，スクリプト化の法則を適当に考えてみました……。

- 命令は“<”と“>”のタグで囲む。
- “*”はラベルの意味
- “$”はコマンド選択の意味
- “#”は画像表示の意味

……この法則にしたがって，「キュートちゃんの冒険」をスクリプト化してみるとリスト1のような文字列データ（char配列）になりました。なんか暗号みたいで見づらいですが，シナリオの修正には便利です。あとは，このデータを解読するプログラムを作らなくてはいけませんが，ここで必要になるのは文字列操作の技術です。たとえば，文字列データからタグを抽出する方法ですが，具体的にはリスト2のような処理を行っています。

《リスト1》「キュートちゃんの冒険」のシナリオをスクリプト化したプログラム・リスト

```
/*シナリオデータ*/
char cSina[]={
    "<*最初>キュートちゃんの冒険<#gazou1.bmp><$柿=ゲームスタート>"
    "<*柿>カキがたおれている<#gazou2.bmp>"
    "<$クリア=ふみつぶす,死=たすける>"
    "<*死>柿に襲われた<#gazou3.bmp><$最初=最初に戻る>"
    "<*クリア>おめでとう<#gazou4.bmp><$最初=最初に戻る>"
};
```

《リスト2》タグで囲まれた特定の文字を検索する関数

```
/*----------------------------------------*/
/*“<*ラベル名>”タグの文字列を検索する*/
/* 引数（検索元のシナリオ，検索する文字列） */
char * fncGoto(char *pSina ,char *pLabel)
{
    char *pTag;

    for(;;) {
        for(;;) {
            /*"<"タグ待ち*/
            if(*pSina++=='<') {
                pTag = &cTag[0]; /*タグ初期化*/
                break;
            }
        }
        for(;;) {
            /*">"タグ検出中*/
            if(*pSina=='>') {
                *pTag = NULL; /*タグ格納完了*/
                if((cTag[0] == '*')&&(fncStrcmp(&cTag[1] ,pLabel)==TRUE))
                    return(pSina+1);  /*“<*ラベル名>”発見！*/

                break;  /*シーケンス移行*/
            } else {
                *pTag++ = *pSina++; /*タグ格納中*/
            }
        }
    }
    return(NULL);
}
```

■シナリオの拡張も可能

これらをさらに発展させて，スクリプト部分をテキスト・ファイル化してしまえば，シナリオを後から拡張できるようになります。他にも，フラグを持たせたり，フラグによって分岐する機能をつけてしまえば，かなり出来のいいアドベンチャー・ゲームのシステムになるはずです。

今回はシステムのみで，ストーリーは前回のまま……にしようかと思ったのですが，それだと寂しいので，「キュートちゃんの冒険2」というサンプルを作ってみました。内容は遊んでみてからのお楽しみです。

プログラムはマイコンBASICマガジンのWEBサイトからダウンロードすることができます。

http://www.basicmagazine.net/

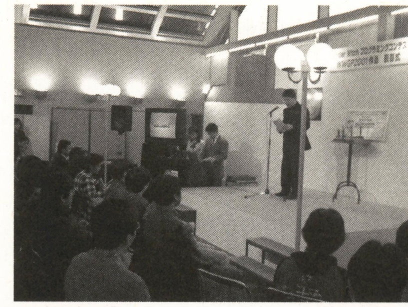




▲サンプルに作ったアドベンチャー・ゲーム「キュートちゃんの冒険2」

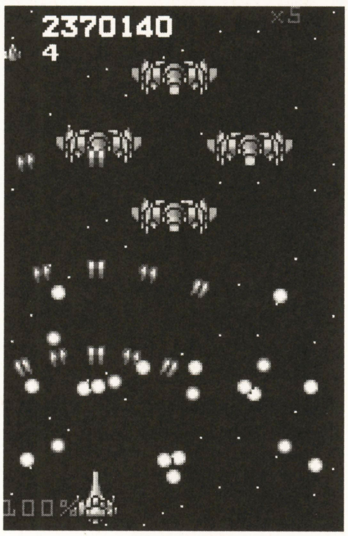
WWGP2001 最終選考会!!

3月3日，ワンダーウィッチによるプログラムコンテスト「WWGP2001」の最終選考会が行われ，遂にグランプリが決定しました。果たして，全147作品の頂点に立ったのは一体どの作品でしょうか。



グランプリ（賞金50万円ほか）

「JUDGEMENT SILVERSWORD」
（ジャッジメント シルバーソード）（カラー専用）
作者：M-KAI（えむかい）さん

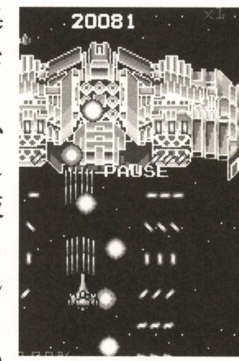


▶気持ちいいくらい弾が乱れ飛ぶ、超本格的シューティング・ゲーム

縦画面を使ったシューティング・ゲームです。雨あられと降りそそぐ敵弾。テンポのいいBGM。凝った敵の動き。ド派手なエフェクトなど，作者が「爽快感を重視した」というだけあって遊んで楽しさが感じられる見事な作りになっています。

自機は移動のほかに「前方ショット」「フォースフィールド(バリア)」「ワイドショット」という3種類のアクションが可能です。敵弾はフォースフィールドで防げますが（耐久力に限界あり）その間は攻撃ができません。ワイドショットは広範囲に攻撃できますが，連射性が前方ショットよりも劣ります。このため，プレイヤーは状況によって攻撃を使いわける必要があり，奥深いゲームになっています。

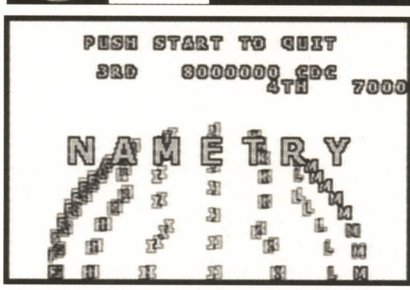
審査員の方からは「もう少しがんばれば市販品に負けないものになる」「遊んで爽快感があって楽しい」とのコメントがありました。



▲13面に登場する巨大キャラ。戦いはまだまだ続く（全30面！）

準グランプリ（賞金10万円）

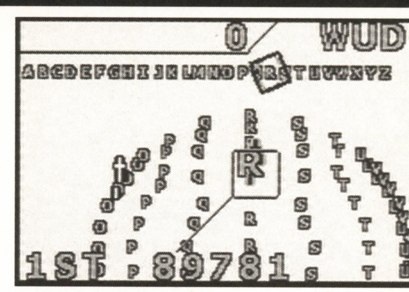
「なめとらい」（カラー／モノクロ両対応）
作者：長 健太（ABA."Saba"）さん



▲文字が球状にグルグルと動き回るのが気持ちいい

アーケード・ゲームなどでおなじみのスコアランキングの名前入力をゲームにしてしまったのが，この「なめとらい」です。

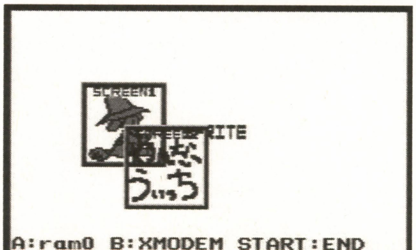
作者さんの「こんなのもアリ」という言葉どおり，かなり突飛なゲーム内容ですが，球面に配置された文字がスクロールするさまは，見ていて気持ちが良く，技術的にも大変評価されていました。



▲"十"の姿をした敵をかわして，文字を入力！

ツール賞（賞金5万円ほか）

「画面取り込みライブラリ BMPSaver ver2.01」
（カラー／モノクロ両対応）
作者：Kazuhide Hatsuyama（WakuWaku）さん

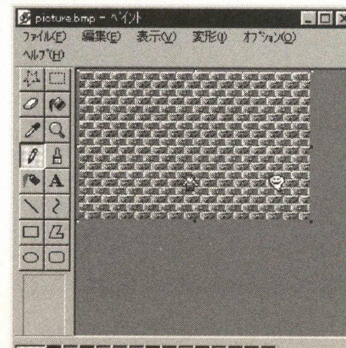


▲画面取り込みライブラリを組み込んだサンプルプログラム。ボタン操作でキャプチャできます

審査員に「ほとんどのワンダーウィッチユーザーがお世話になっている」と言わしめる，定番キャプチャーツール。

ワンダーウィッチで実行中のスクリーンショットを画像ファイルとして生成することができる便利なツールです。

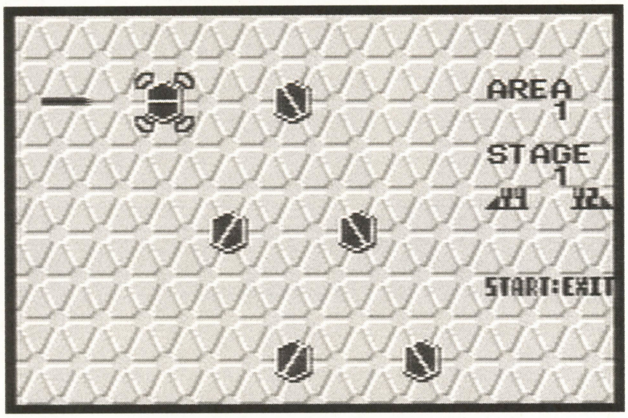
最新バージョンではXMODEMによる画像ファイルの転送や，カラー対応などの機能追加が行われています。



◀見事、パソコンに表示されたスクリーンショット

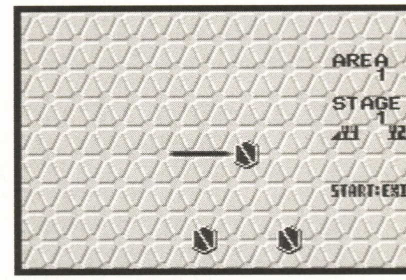
アイデア賞（賞金5万円ほか）　「石　狩」（モノクロ作品）

作者：花井章能さん



▶反射を考えて，石の向きを設定中

非リアルタイム型のパズルゲームです。ルールは石に光線を当てて消していくというものですが，光線は石に当たると反射します。石は60度単位で回転させることができます。一発勝負で石を消していくのは，さながらドミノ倒しのような快感があります。

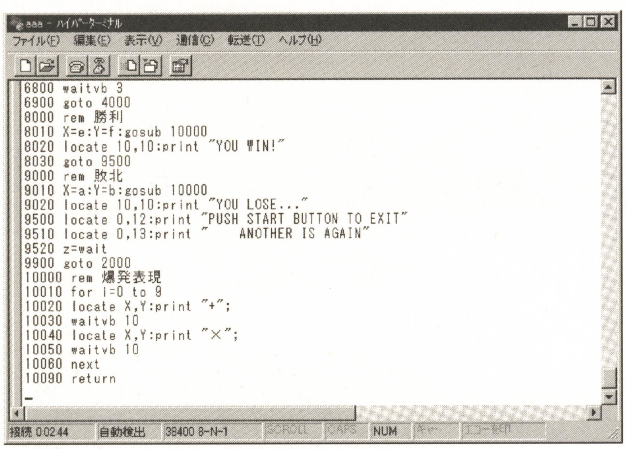


▲一発勝負で光線を発射!!

テクニカル賞（賞金5万円ほか）　「ワンべぇ：WONBEWonderWitch BASIC Environment」（カラー／モノクロ両対応）

作者：株式会社ピーデーの川俣晶さん(Play文機能に関してHirotaka JOE Ohkuboさん)

「ワンべぇ」はワンダーウィッチでBASICが動いてしまうという驚きのソフトウエアです。搭載されたBASICインタプリタによって，BASIC言語で書かれたプログラム（拡張子“wb”）を実行することができます。さらに，スワンケーブルでパソコンとワンダースワンを接続してから，通信ソフト（ハイパーターミナルなど）を立ち上げれば，パソコンからの入力でプログラミング&実行を簡単に行うことができます。

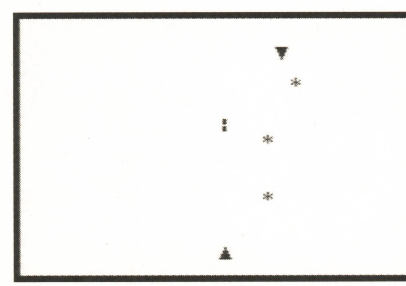



```
6800 waitvb 3
6900 goto 4000
8000 rem 勝利
8010 X=e:Y=f:gosub 10000
8020 locate 10,10:print "YOU WIN!"
8030 goto 9500
9000 rem 敗北
9010 X=a:Y=b:gosub 10000
9020 locate 10,10:print "YOU LOSE..."
9500 locate 0,12:print "PUSH START BUTTON TO EXIT"
9510 locate 0,13:print "    ANOTHER IS AGAIN"
9520 z=wait
9900 goto 2000
10000 rem 爆発表現
10010 for I=0 to 9
10020 locate X,Y:print "+";
10030 waitvb 10
10040 locate X,Y:print "×";
10050 waitvb 10
10060 next
10090 return
_
```

▲通信ソフトを使ってプログラミングもできる

「ワンべぇ」の言語仕様は“color”“locate”“print”など，「Tiny BASIC」に似た形式で，“peek”“poke”“play”文にも対応しているという念の入れようです（なお，作者の川俣さんは会社名でエントリーされていますが，「ワンべぇ」は個人的な趣味として作られたものだそうです。すごい）。

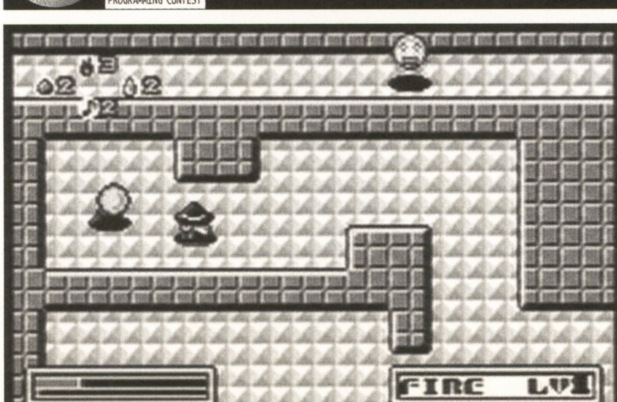
今回の受賞では，BASICによってプログラミングに対するしきいを下げた功績が評価されたようです。「ワンダーウィッチを使ってみたいけどC言語はむずかしい」という人はゼヒ使ってみてはいかがでしょうか？



▲サンプル“shoot.wb”の実行結果。BASICなのでプログラムの途中停止も可能

エンターテイメント賞（賞金5万円ほか）　「MagicalHarvest」（カラー／モノクロ両対応）

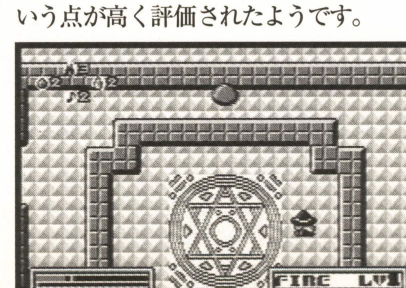
作者：紫雨陽樹さん



▲全128もの広大なマップを持つ本格的なアクションRPG

全8ステージ（128マップ）の本格的なアクションRPGです。主人公を操って敵を倒し，マップを攻略していきます。魔導書や石版を手に入れると魔法が使えるようになります。魔法を使うとマナを消費しますので，敵を倒して補充することができます。そして，ステージごとにボスが登場します。

「マップが多くて，敵の動きもいい」という点が高く評価されたようです。



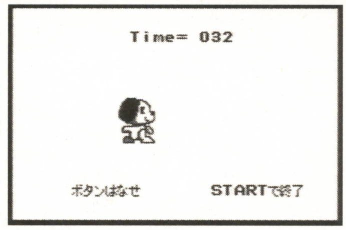
▲ここであることをするとゲートが開いてボスキャラとの戦闘に突入するらしい

審査員特別賞（賞金5万円ほか）

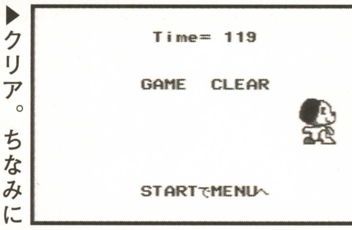
「いぬ　れんだ」（モノクロ作品）
作者：BlueSkyさん

この「いぬ　れんだ」，審査員側から「なにか足りないのがいい」という評価を得て特別賞に輝きました。その名のとおり，ひたすらボタンを連打して犬のキャラクターを前進させてタイムを競うというピコピコ・ゲームです。

▶ボタン連打中。ちなみに足は動かない



▶クリア。ちなみにゴールラインはない

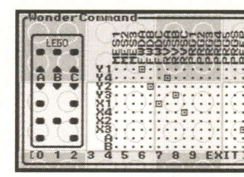


佳作（WonderSwanの貴重な外装をプレゼント）

「WonderCommand（ワンダーコマンド）」（カラー／モノクロ両対応）
作者：ななしのさん

「WonderCommand」はLEGO MINDSTORMS「Robotics Invention System（RIS）」のRCX（マイコン内蔵ブロック）に赤外線メッセージを送信するソフトウエアです。RCXにモータを接続しておけば，ワンダースワンからのボタン操作で，そのモータを回転させることができます。たとえば，MINDSTORMSで車を作ることでラジコンカーのように操作して遊べます。

「WonderCommand」を楽しむためには，自作ハードウエアである「Wonder Command-Ⅱ」が必要です。「Wonder Command-Ⅱ」はAVR（マイクロコントローラ）と赤外線LEDで構成されていて，外装とコネクタ部分は「WonderWave」を流用していました。

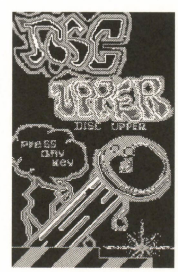


▶「Wonder Command」の基本画面

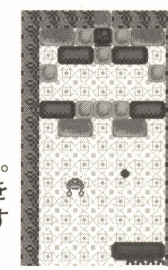
「Disc Upper」（カラー専用）
作者：沖 賢樹さん

独自の操作系を持ったブロックくずし風ゲームです。二つの方向キーのうち左側でUFOを，右型でバー（パドル）を動かします。バウンドするボールはバーで跳ね返すことができますが，UFOに当たるとミスになってしまいます。

ワンダースワンの特徴を生かした縦画面や，二つある方向キーを使ってた点が評価されたようです。



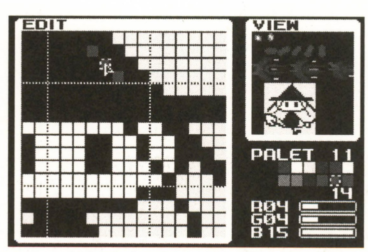
◀「Disc Upper」オープニング画面



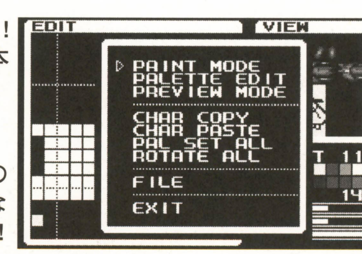
▶ゲーム中の画面。二つの方向キーを駆使してプレイするのが新しい

「Yuuki! Paint for Wonder Witch」（カラー専用）
作者：米沢勇気（Yuuki!）さん

ワンダースワンカラーを使って128×128ピクセル，最大241色のドット絵を描くことができるツールです。動作環境が実機であるワンダースワンカラーであるため，いつでもどこでもドット絵が描けます（パレットは16色と4色モード両方に対応）。作成した画像データは，独自形式ファイル（拡張子が“ypi”），リソースファイル（拡張子が“fr”），パレットファイル（拡張子が“ypa”）の3種類で保存できます。



◀「Yuuki! Paint」の基本画面



▶パレットのエディットなど機能も豊富!

◆最後に

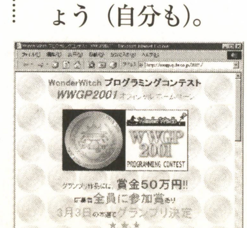
「ワンダーウィッチでここまでデキるのか！」と新鮮な驚きの連続だった「WWGP2001」が遂に終了しました。

この「WWGP2001」，本選で作者自らが作品のプレゼンテーションを行ったり，応募作品をWEBサイトで公開してしまったりと，数多くの画期的な試みがなされていました。応募作品は審査員1人1人が全部目を通したとのことですが，その選考は大変だったようです。本選でも，甲乙つけがたい作品が多く，急きょ「特別審査員賞」や「佳作」が増える結果となりました（審査員のみなさまもご苦労様でした）。すでにホビープログラミングのすそ野が広がっている証拠でしょうか。

では次回の「WWGP2002（仮）」までにプログラミングの腕を磨いておきましょう（自分も）。



▶予選を勝ち抜いた20組全員で記念撮影



「WWGP2001公式サイト」
http://wwgp.qute.co.jp/2001/
今回応募された全作品をダウンロードして楽しむことができます（一部例外あり）。

Catlikeの

もっと身近にプログラミング！

第4話

タイピング・ゲーム編　その1　カナ入力とローマ字入力

日本語をプログラムでとらえると，コンピュータで扱うときはなんて不便なんだろうって思います。あなたは，文章を打つとき，カナ入力派？　それともローマ字派？

筆者紹介

catlikeホームページ
http://catlike.marchen.to/
発表が近づいている新企画。「めるへんうぉーかー」以外のcatlikeタイトルがついに立ち上がります。専用ホームページも準備中。来月号時には「大公開!」されているはずなので,最新情報はホームページでチェックだっ!

文:まかべひろし
新企画のことを書きたいけど,ここはプライベートのことなので別のことを…。NEOGEO64の基板を買いました。ポリゴン侍魂を遊ぶため。最近は基板も値段が落ちたようで,発売時から2桁値段が違ってびっくり。でも,自宅の14インチテレビでは画面が見づらく,大きなテレビが置ける場所がほしいなぁと思ってたりします。

絵:桜部のとす
花粉症がヒドくて大変です…なんだか去年あたりから咳が出てツライのですよ〜。作業がはかどらなくて困ります,とほん。早く花粉のない季節になれ〜。

今年は連休，少ないかも

土曜日はお休み，というのが一般的になってから，祝日が土曜日に入ってくるのがもったいなく思うようになりました。昨年のゴールデンウィークは，うまくこの隙間に入って，あわよくば10日間という大連休になりましたね。それに比べて今年のそれは，なんだか祝日が無駄に消費されている気がします。

昨年の体育の日の例もありますし，ゴールデンウィークもうまいことずれるとうれしいのですけれどね。なんだかんだ言っても，やっぱり長期連休というのはウキウキするものです。

新しいジャンルはどう？

相変わらず関係ない時期ネタで入りましたが，発売日的には新学期ですね。本連載もついについに，新しい話に入ります。

トランプ・ゲーム，アドベンチャー，アクションと続いて，さて次はと考えたのですが，ふと思うと，最近のゲームは昔と比べて多くのジャンルにわたっています。順当ならば，シューティングやシミュレーション，そして誰もが一度は組みたくなるRPGと流れていくのだと思いますが，ひとつ，あまり扱われていないジャンルに思い当たりました。

それは，最近多く見受けられるようになった，タイピング・ゲームというジャンル。有名どころでは，アーケードで人気を博した『TYPING OF THE DEAD』や，「ゥアタタタタタッ」で有名な『北斗の拳』などがあります。コンシューマにはキーボードが標準搭載でないため，ほとんど見受けられませんが，パソコン・ソフトでは結構な市場を確保しているようです。ゲームを遊ぶことでキー入力が速くなるということもあり，コンピュータ初心者から中級者に人気が高く，ゲームソフトの売り上げランキングに顔を出すほどになっています。

しかも，打ち込むだけというプロセスなのに，ゲームとしてなかなか面白かったり。ただキーボードで文章をなぞることが，意外に燃えさせてくれるのです。

システムとして考えると，キャラクターやストーリーを無理なく乗せることができ，しかも文章を打ち込ませることによって，読み流されやすいシナリオを印象付ける手法として，タイピング・ゲームはもっと面白くなるはずとボクは確信しています。アクションやアドベンチャーのように，もっともっとこのシステムに工夫を凝らしたゲームが出てきても良いのでは！

そんなわけで今回は，ちょっと小粒だけど魅惑なテーマ「タイピング・ゲーム」を取り上げてみたいと思います。

文字入力方法

タイピング・ゲーム，これが英語の練習ソフトなら話は簡単なのですけれど，日本語を打ち込むときはやっかいなことに，カナ打ちとローマ字打ちの2タイプの入力方法があります。カナ打ちは１キーあたり１文字ですから，比較的簡単にプログラム処理をすることができるように思えますが，ローマ字打ちの場合，「し」ひとつ取ってみても「Si」「Shi」と2種類のキー入力方法があり，これらを全部処理することを考えると，結構途方に暮れてしまいます。

ゲーム作りの基本は「何でもできるこ

と」ではなく,「いかに機能をそぎ落としてスマートな仕様にまとめるか」であるとは思うのですけれど，だからと言ってタイピング・ゲームで「カナ打ちオンリーですからローマ字打ちの人はご遠慮くださぁい♪」と言い切るのは，さすがに抵抗ありますよね。自分がカナ打ちの人は「そうか？」と思うかもしれませんけれど，少なくともボクはローマ字打ちだったりして，自分の遊べないゲームを作るのはやっぱり面白くはないものです。

ちなみに，まともに統計を取ったわけではないのですけれど，日本人の大多数はローマ字打ち入力のような気がします。気がするだけかもしれませんけど。

そんなわけで，せめて「ローマ字打ち版」と「カナ打ち版」を作れるように，できればオプションでそれを切り替えたいところです。ですが，このまったく違う2系統の入力方法をスムーズにプログラムするには，どのようにすればよいのでしょうか。

今回の例のように，似て異なるプログラムを統合するには，いくつかのアプローチがあります。まったく違うそれぞれのプログラムを組んでしまうのが一番手っ取り早いと思います。一つのプログラムでどちらの機能も実現する場合，よっぽどそれぞれの組みかたがまとまっていない限り，きれいに統合されません。また，その切り替え部分がバグの要因になったりします。

簡単! 単純! カナ打ちルーチン

さて，ちょっと話が脱線してしまいましたけれど，文字の入力方法。「ローマ字入力」と「カナ入力」，どちらがプロ



グラムを組むのが簡単か，と言われればカナ入力のほうが簡単でしょう。そうイメージさせる理由は，やはりカナ文字の場合，入力方法が多岐にわたっていないためだと思います。

また，キー一つに対して1文字と割り当てられているので，処理は比較的簡単です。入力されたキーに対応する文字と処理すればよいだけです。

ただし，カナ入力には，ほんのちょっとだけ特殊な文字があります。「゛」「゜」,「ぁ」や「っ」などのSHIFTキーとの組み合わせで打つ文字は，前述した「キー一つ=1文字」のルールが当てはまりません。プログラムに特例というのは，アルゴリズムという点ではきれいではありませんが，この程度であれば特例処理を行ってもたいした量ではありません。

「゛」「゜」が押された場合，ひとつ前に処理した文字を見て,「゛」もしくは「゜」がつけられる文字なら，ひとつ前の文字を変更します。

SHIFTキーとの組み合わせで打つ「ぁ」「っ」等の小文字は，他の文字と同じように処理し，SHIFTキーが押されていて，かつ組み合わされた文字に小文字が存在する場合は，小文字を打ち込んだものとし，そうでなければSHIFTキーは無視という形で処理をしましょう。

そう単純に考えてプログラムを組むと，**リスト1**のような感じになると思います。やることは，キー入力をとって，それに対応した文字列を返すというだけですね。

物量は多いですが，一つ一つは非常に単純なコードです。カナ五十音分，正確には小文字も入るのでもっと多くなりま

《リスト1》キー入力に対応した文字列を返すサンプル・プログラム

```
-----------------------------------------------------------
char *keypunch() {
    if ( GetAsyncKeyState( '3' ) ) {
        if ( GetAsyncKeyState( VK_SHIFT ) ){ return "ぁ"; }
        return "あ";
    }
                :
                :
                :
    if ( GetAsyncKeyState( 'Y' ) ) { return "ん"; }
    if ( GetAsyncKeyState( '@' ) ) { return -1; }
    if ( GetAsyncKeyState( '[' ) ) { return -2; }

    return NULL;
}

void sample() {
    char szString[100];
    strcpy( szString, "" );
    while( 1 ) {
        char *p = keypunch();
        if ( p == -1 ) {
            // 濁点が入った場合
            int i,j;
            char tbl[] = "かきくけこさしすせそたちつてとはひふへほ";
            char xxx[] = "がぎぐげござじずぜぞだぢづでどばびぶべぼ";
            j = strlen( szString );
            if ( j > 0 ) {
                for ( i = 0; i < strlen( tbl ); i+=2 ) {
                    if ( szString[j-2] == tbl[i] && szString[j-1] == tbl[i+1] ) {
                        szString[j-2] = xxx[i]; szString[j-1] = xxx[i+1];
                    }
                }
            }
        } else if ( p == -2 ) {
            // 半濁点が入った場合
            int i,j;
            char tbl[] = "はひふへほ";
            char xxx[] = "ぱぴぷぺぽ";
            j = strlen( szString );
            if ( j > 0 ) {
                for ( i = 0; i < strlen( tbl ); i+=2 ) {
                    if ( szString[j-2] == tbl[i] && szString[j-1] == tbl[i+1] ) {
                        szString[j-2] = xxx[i]; szString[j-1] = xxx[i+1];
                    }
                }
            }
        } else if ( p != NULL ) {
            strcat( szString, p );
        }
    }
}

※GetAsyncKeyState()はWindowsAPIのひとつ。
  現在，指定したキーが押下されているかを調べることができる。

-----------------------------------------------------------
```

すが，その分だけプログラムを記述することで，カナ打ちのルーチンを作成することができます。

ちょっと特殊なのは，濁音，半濁音の部分ですが，この程度の特例処理は「特殊」という感じもしないと思います。sample()の文字判定のところで2byte分調べている理由は，日本語は2byte文字だからです。

ローマ字入力は大量だよ

問題はもう一方，ローマ字入力のほうです。カナ打ちと同様，大量なコードで処理をしてもいいのですが，一番の問題になるのは，ローマ字の場合は1キー1文字というわけではないところです。ひとつの文字「し」を打ち込むにしても，「SI」と2文字打つこともありますし「SHI」と3文字打つこともあります。

この例でしたら，以下のようにテーブルを持ちます。

```
char p[][2][8] = {
      { "し", "SI" },
      { "し", "SHI" }
};
```

p[0][0]およびp[1][0]には「し」，p[0][1]には「SHI」，p[1][1]には「SI」が入ります。[x][0]は表示される文字，[x][1]はキーボードから入力されるデータと考えてください。この形式でしたら，ただ単純にデータを増やしていけば，反応する文字も増えていきます。

このテーブルをもとに判定をしていきます。基本的には，[x][1]のほうのデータにキーボードから入力された文字を照らし合わせていき，一致したものの[x][0]を入力されたデータとして扱います。

タイピング・ゲームの場合，ほとんどは「打ち込ませたい文字列」というのが存在するので，そちらと[x][0]を参照して「打ち込めるデータ」をあらかじめ選定しておくことが大事になります。フリーでどんな文字でも打てるとなると，タイピング・ゲームではなく，ワープロ・ソフトを作ることになっちゃいますしね。そういうゲームもアリかもしれませんけど，今回の趣旨ではないので無視しちゃいますね。

《リスト2》ローマ字入力すべてテーブルに入れたサンプル・プログラム

```
---------------------------------------------------------

void sample() {
    // 問題文
    char szTarget[] = "しま";

    // ローマ字テーブル(本来は全文字分ある)
    char p[][2][8] = {
        { "し", "SI" },
        { "し", "SHI" },
        { "ま", "MA" }
    };
    // ローマ字テーブル用フラグ
    int n[sizeof( p ) / (8 * 2)];
    // 問題文読み進めカウンタ
    int nCnt;
    // ローマ字読み進めカウンタ
    int nPnt;

    // 問題文字数分ループします
    for ( nCnt = 0; nCnt < strlen( szTarget ); nCnt += 2 ) {
        // 問題文から，ローマ字テーブルフラグを初期化
        int i,nEndFlg;
        char c[3];
        c[0] = szTarget[nCnt]; c[1] = szTarget[nCnt+1]; c[2] = 0;
        for ( i = 0; i < sizeof( p ) / (8 * 2); i++ ) {
            if ( strcmp( p[i][0], c ) == 0 ) { n[i] = 1; }
            else { n[i] = 0; }
        }

        // 入力判定
        nPnt = 0; nEndFlg = 1;
        while( nEndFlg ) {
            char k = getch();
            for ( i = 0; i < sizeof( p ) / (8 * 2); i++ ) {
                if ( n[i] == 1 ) {
                    if ( p[i][1][nPnt] == k ) {
                        // 入力したキーはローマ字テーブルに一致
                        nPnt++;
                        if ( p[i][1][nPnt] == 0 ) {
                            // 入力一文字終了！
                            nEndFlg = 0;
                        }
                    } else {
                        // 入力したキーはローマ字テーブルから外れた
                        n[i] = 0;
                    }
                }
            }
        }
    }
    // 入力終了したらここに来ます
}

---------------------------------------------------------
```

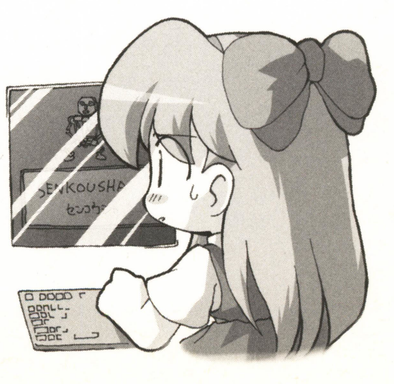
実際にローマ字入力としてあるものをすべてテーブルに入れると，そのテーブルは膨大なものになりますが，プログラムは意外とすっきりしたものになります（**リスト2**）。サンプルのほうは，かなり単純化していますが，量も少ないのでプログラムも追いやすいと思います。

チェック・ポイントは，テーブルの各項目に対応したフラグを持っていること。サンプルでは「しま」という謎な文章を打ち込んでもらうプログラムになっていますが，その1文字ごとにループで処理をしています。

最初に「し」を打ち込ませるために，ローマ字テーブルフラグを初期化します。テーブルの中で「し」を打ち込めるのは，p[0]の「し，SI」とp[1]の「し，SHI」です。この二つに対応したフラグn[0],n[1]に1を入れ，対応していないp[2]の「ま，MA」に対応したフラグn[2]には0を入れます。このフラグに1が入っているテーブルのみを処理していきます。

getch()は，キーボードから1文字入力を得るC言語標準の関数です。何かキーが押されるまで待ち，押された結果をkに入れます。

そして，ローマ字テーブルフラグに1が入っているテーブルのp[x][1]と入力



結果kを比べ，同じならばp[x][1]を読み進めていき，最後まで読み進むことができたら，その文字の入力が成立したということになります。

1文字成立したら，つぎは「しま」の2文字目,「ま」の処理を同様に行います。全文字入力が終了したら，このプログラムは終了します。

流れはさほど複雑ではありませんが，ローマ字テーブルを全文字分記述すると，szTargetに打ち込ませたい文字を入れてこのプログラムを呼ぶだけで，ローマ字入力の判定プログラムが完成するわけです。

まぁ，このままゲームに使うとなると，画面演出などまったくないのでお話にはなりませんが，流れさえ理解すれば，自分で同様のルーチンを書くのも難しくはないと思います。

次回もあるよ

タイピング・ゲームのキモとなる，入力部分の紹介は，簡単ですが終わってしまいました。今回の「タイピング・ゲーム編」，もう終わりじゃない？　と思ってしまうところですが，なんの，その2のネタもちゃんとあります。

たとえば，ローマ字入力。「ん」の入力方法を考えてみてください。「N」と答える人もいれば「NN」という人もいると思います。知ってのとおり，どちらも間違ってはいません。今回のプログラムでは，この両方に正常に対応することができません。

また，画面の演出を入れると，演出の待ち時間のおかげで途端に入力反応が悪くなってしまうケースがあります。DOS時代のプログラムなら，これはさほど起こらなかったのですが，Windows時代のプログラム，とくにDirectInputを使ったケースの場合，うっかりすると間違った組みかたをしてしまう可能性があります。

そしてそして，普通は扱わないケース。ローマ字とカナの同時入力方法というものについても考えてみたいと思います。市販されているゲームは，いくつかを除いて，オプションでローマ字入力とカナ入力を変更したり，それぞれを別パッケージで売り出したりしています。では，なぜ双方の同時入力ができないのか。同時入力をさせる方法はないのか。じつは最近，同時入力をサポートした市販ゲームのプログラムをやりましたので，引っかかった部分のお話もできると思います。

ほら，タイピング・ゲーム，奥深そうでしょう？　まだまだこのジャンルのソフトは，他のジャンルに比べれば少ないですし，アイデアしだいで面白いゲームが作れると期待しています。ぜひ，このインターフェースで，面白いゲームを作り出してください。

《第1表》ローマ字表。いろんな記述方法があるんだよね

文字	カナ	ローマ字		
あ	3	A		
い	E	I		
う	4	U		
え	5	E		
お	6	O		
ぁ	3	LA	XA	
ぃ	E	LI	XI	
ぅ	4	LU	XU	
ぇ	5	LE	XE	
ぉ	6	LO	XO	
か	T	KA		
き	G	KI		
く	H	KU		
け	[ba]	KE		
こ	B	KO		
きゃ	G7	KYA		
きぃ	GE	KYI		
きゅ	G8	KYU		
きぇ	G5	KYE		
きょ	G9	KYO		
くぁ	G3	QWA	KWA	
さ	X	SA		
し	D	SI	SHI	
す	R	SU		
せ	P	SE		
そ	C	SO		
しゃ	D7	SYA	SHA	
しぃ	DE	SYI		
しゅ	D8	SYU	SHU	
しぇ	D5	SYE	SHE	
しょ	D9	SYO	SHO	
た	Q	TA		
ち	A	TI	CHI	
つ	Z	TU	TSU	
て	W	TE		
と	S	TO		
っ	Z	LTU	XTU	
ちゃ	A7	TYA	CYA	CHA
ちぃ	AE	TYI	CYI	
ちゅ	A8	TYU	CYU	CHU
ちぇ	A5	TYE	CYE	CHE
ちょ	A9	TYO	CYO	CHO
つぁ	Z3	TSA		
つぃ	ZE	TSI		
つぇ	Z5	TSE		
つぉ	Z6	TSO		
てゃ	W7	THA		
てぃ	WE	THI		
てゅ	W8	THU		
てぇ	W5	THE		
てょ	W9	THO		
とぁ	S3	TWA		
とぃ	SE	TWI		
とぅ	S4	TWU		
とぇ	S5	TWE		
とぉ	S6	TWO		
な	U	NA		
に	I	NI		
ぬ	1	NU		
ね	[bc]	NE		
の	K	NO		
にゃ	I7	NYA		
にぃ	IE	NYI		
にゅ	I8	NYU		
にぇ	I5	NYE		
にょ	I9	NYO		
は	F	HA		
ひ	V	HI		
ふ	2	HU	FU	
へ	[de]	HE		
ほ	[bd]	HO		
ひゃ	V7	HYA		
ひぃ	VE	HYI		
ひゅ	V8	HYU		
ひぇ	V5	HYE		
ひょ	V9	HYO		
ふぁ	23	FA		
ふぃ	2E	FI		
ふぇ	25	FE		
ふぉ	26	FO		
ふゃ	27	FYA		
ふぃ	2E	FYI		
ふゅ	28	FYU		
ふぇ	25	FYE		
ふょ	29	FYO		
ま	J	MA		
み	N	MI		
む	[dd]	MU		
め	[bf]	ME		
も	M	MO		
みゃ	N7	MYA		
みぃ	NE	MYI		
みゅ	N8	MYU		
みぇ	N5	MYE		
みょ	N9	MYO		
や	7	YA		
い	E	YI		
ゆ	8	YU		
いぇ	E5	YE		
よ	9	YO		
ゃ	7	LYA	XYA	
ぃ	E	LYI	XYI	
ゅ	8	LYU	XYU	
ぇ	5	LYE	XYE	
ょ	9	LYO	XYO	
ら	O	RA		
り	L	RI		
る	[be]	RU		
れ	[bb]	RE		
ろ	[e2]	RO		
りゃ	L	RYA		
りぃ	LE	RYI		
りゅ	L8	RYU		
りぇ	L5	RYE		
りょ	L9	RYO		
わ	0	WA		
うぃ	4E	WI		
う	4	WU		
うぇ	45	WE		
を	0	WO		
ん	Y	N		
が	T[c0]	GA		
ぎ	G[c0]	GI		
ぐ	H[c0]	GU		
げ	[ba][c0]	GE		
ご	B[c0]	GO		
ぎゃ	G[c0]7	GYA		
ぎぃ	G[c0]E	GYI		
ぎゅ	G[c0]8	GYU		
ぎぇ	G[c0]5	GYE		
ぎょ	G[c0]9	GYO		
ぐぁ	H[c0]3	GWA		
ぐぃ	H[c0]E	GWI		
ぐぅ	H[c0]4	GWU		
ぐぇ	H[c0]5	GWE		
ぐぉ	H[c0]6	GWO		
ざ	X[c0]	ZA		
じ	D[c0]	ZI	JI	
ず	R[c0]	ZU		
ぜ	P[c0]	ZE		
ぞ	C[c0]	ZO		
じゃ	D[c0]7	JYA	ZYA	JA
じぃ	D[c0]E	JYI	ZYI	
じゅ	D[c0]8	JYU	ZYU	JU
じぇ	D[c0]5	JYE	ZYE	JE
じょ	D[c0]9	JYO	ZYO	JO
だ	Q[c0]	DA		
ぢ	A[c0]	DI		
づ	Z[c0]	DU		
で	W[c0]	DE		
ど	S[c0]	DO		
ぢゃ	A[c0]7	DYA		
ぢぃ	A[c0]E	DYI		
ぢゅ	A[c0]8	DYU		
ぢぇ	A[c0]5	DYE		
ぢょ	A[c0]9	DYO		
でゃ	W[c0]7	DHA		
でぃ	W[c0]E	DHI		
でゅ	W[c0]8	DHU		
でぇ	W[c0]5	DHE		
でょ	W[c0]9	DHO		
どぁ	S[c0]3	DWA		
どぃ	S[c0]E	DWI		
どぅ	S[c0]4	DWU		
どぇ	S[c0]5	DWE		
どぉ	S[c0]6	DWO		
ば	F[c0]	BA		
び	V[c0]	BI		
ぶ	2[c0]	BU		
べ	[de][c0]	BE		
ぼ	[bd][c0]	BO		
びゃ	V[c0]7	BYA		
びぃ	V[c0]E	BYI		
びゅ	V[c0]8	BYU		
びぇ	V[c0]5	BYE		
びょ	V[c0]9	BYO		
ぱ	F[db]	PA		
ぴ	V[db]	PI		
ぷ	2[db]	PU		
ぺ	[de][db]	PE		
ぽ	[bd][db]	PO		
ぴゃ	V[db]7	PYA		
ぴぃ	V[db]E	PYI		
ぴゅ	V[db]8	PYU		
ぴぇ	V[db]5	PYE		
ぴょ	V[db]9	PYO		
ヴぁ	4[c0]3	VA		
ヴぃ	4[c0]E	VI		
ヴ	4[c0]	VU		
ヴぇ	4[c0]5	VE		
ヴぉ	4[c0]6	VO		
ヴゃ	4[c0]7	VYA		
ヴぃ	4[c0]E	VYI		
ヴゅ	4[c0]8	VYU		
ヴぇ	4[c0]5	VYE		
ー	[dc]	[bd]		

※キーボードコードは、WindowsのDirectInputのもの
※[xx]は、16進数のキーボードコード。

catlike最新情報

先月ちょろっとお話ししていた新企画ですが，公開できる日が近づいてきました。今回は，スタッフに雰囲気を掴んでもらうため作ったイメージ画像を紹介。絵師，桜部のとすのかわいいキャラクターにマッチするよう，ふきだしによるメッセージ表示システムを採用し，文章でも演出するアドベンチャー・ゲームに仕上がってきています。

ゲームの入手方法なのですが，catlikeのホームページにて，毎月1話ずつダウンロードしていただけます。そして，プレイヤーのみなさんと一緒にこのゲームを盛り上げていけたらいいな，と思っています。もちろんダウンロードは無料です！

インターネット以外でも，どうにかみなさんにゲームをお届けできないか，と方法を模索中です。いい方法がありましたら，アンケートはがきにでも書いて教えてください。

とにもかくにも。ダウンロード開始は4月下旬。専用ホームページも立ち上げますので，ぜひぜひチェックしにきてくださいね！



▲画面は超開発中です。レイアウトや雰囲気はともかく，画像は絶対変わります



初心者でもかんたんラクラク作れちゃう！！

iモード・ホームページ講座

第2回　簡単でちょっと見栄えがよくなるタグ

「今の時代，立ち上げるならｉモードサイトだぜ！」そのココロは「だってこっちのほうが簡単なんだもん」というわけで？　今月も「できることからコツコツと」をモットーに，ｉモードで見られるカッコいいホームページを立ち上げるためにがんばりましょう。２回目の今回はhtml文書を構成する「タグ」と呼ばれる記号について解説します。

TEXT●白木ミホ



東京産の神奈川育ち。ｉモード関連の執筆が多い。先日深夜、自宅のパソコンでネットに接続しようとしたら突然、人の声がッ！「ただ今留守にしております…」って、いきなりモデムが喋り出してました(爆)。こんな夜中にどこ行っとんじゃサーバーはっ?!と、独り空しくワケ分からんことを叫んでいた私。こんなことって…

タグってなんだ？

ｉモードあるいは一般のパソコン上で見ることのできるホームページは，どれもタグという記号を使って書かれたhtml文書ファイルである，という話を前回しました。

html文書にはホームページを構成するための情報が「○○という文を表示しなさい」「□□という名前の絵を表示しなさい」というふうに書かれています。しかし，命令文がもちろんこのまま日本語で書かれているわけではありません。日本語で書ければ苦労はしないのですが，残念ながら現在のテクノロジーではムリな話。そこで，端末が理解できる記号を使って書くことが必要になるのです。その記号がhtmlタグと呼ばれる人工言語なのです。

「タグ」言語はホームページ国の公用語です。しかしいったい全部でいくつあるかといえば，それはもう日々次々続々と新しいものが開発されているので，今や辞典ができるほどにとしか言えません。

文字は数字と記号とアルファベットを使います。アルファベットには大文字小文字の区別はありませんが，一つのhtml文書の中ではどちらかに統一したほうが，作るアナタが見やすいという点でよいでしょう。

レイアウトの基本“
”と“<p>”そして“<center>”

前回の実践で，画面に『こんにちは』と表示するためのhtmlテキストを説明しましたが，そこでhtml文書に必要不可欠なタグがすでに登場していました。中でもいちばん大事なのが“<html>”と“</html>”のペア。これがないとhtml文書として認識されないのです。

続いて“<body>”と“</body>”のペア。これが本文を収める部分です。

今回は長いコンテンツを見やすくきれいに構成するために，レイアウトの基本となる四つのタグを覚えましょう。

●

break（改行）の略で，文章を改行させるタグです。これはペアとなる“/”付きのタグが必要ありません。単体で動作します。

●<p>

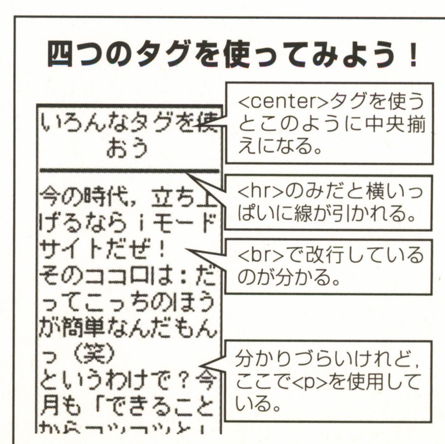
これは段落をつくるためのタグです。“<p>”と“</p>”のセットで使用します。単純に改行するときは“<p>”ではなく，“
”を使用します。

●<center>

“<center>”と“</center>”のセットで

《リスト1》“
”“<p>”“<center>”“<hr>”を使ったサンプルリスト

```
<html>
<head>
<title>表示テスト</title>
</head>
<body>
<center>
いろんなタグを使おう
</center>
<hr>
今の時代，立ち上げるならｉモードサイトだぜ！<br>
そのココロは：だってこっちのほうが簡単なんだもんっ（笑）
<p>
というわけで？今月も「できることからコツコツと」をモットーに，ｉモードで見られるカッコいいホームページを立ち上げるためにがんばりましょう。２回目の今回はhtml文書を構成する「タグ」と呼ばれる記号について解説します。
</p>
</body>
</html>
```



《第1図》これがhtmlタグで作った文書。難しそうに見えて意外と簡単

《リスト2》携帯端末ならではの電話とメールのタグを使ったサンプルリスト

```
<html>
<head>
<title>最近見つけたお店</title>
</head>
<body>
<center>
バラティエール
</center>
<p>
店名はフランス語で「ペテン師」という意味。<br>
別にオーナーがペテン師なんじゃなくって小説のタイトルから取ったらしいよ。<br>
もちろん味はダマシなしの保証つき！
<hr>
電話<A HREF="tel:0123456789">(0123)45-6789</A>
<p>
昼12時～夜9時30分まで
<hr>
「私はこんなステキなお店を知ってる！」という人，下記までメールください。<br>
たかっちへのメールはこちら…<a href="mailto:xxx@xxx.ne.jp">xxx@xxx.ne.jp</a>
</body>
</html>
```

使用します。間に挟んだ文字列を画面中央に揃えて表示させるためのタグです。

●<hr>

horizon（ホライゾン）の略で横線を引くタグ。これも単体で動作します。

このレイアウトをhtmlで書くと**リスト1**のようになり，表示画面は**第1図**です。

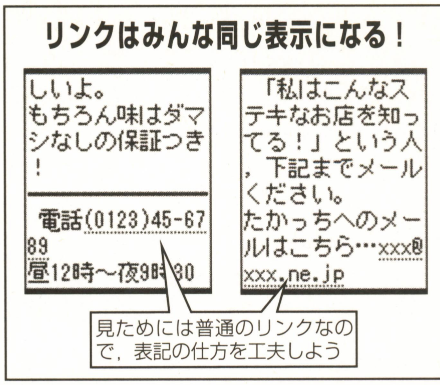
注意！ iモードでは使えないタグ

じつは，私たちがめざすiモード上のホームページ制作にあたって，覚えておかなければならないことがあります。

それはタグというものが「パソコン用サイトでもiモード専用サイトでも使えるもの」，「iモードでは使えないもの」，「iモードでしか使えないもの」の3種類に分かれているということです。

原因は，iモードサイトを構成する言語が一般に使用されるhtml言語を簡略化した「コンパクトhtml」と呼ばれる言語を使用していること，そして，“<font color>”のように文字のカラー指定をするタグなど，機種によっては意味をなさないタグもあるなどさまざまです。iモードの機種向上に従い「使えるタグ」の種類は飛躍的に増えていますが，製作者としては下位機種ユーザーにも優しい画面作りを心がけたいものです。

それになんといってもiモードサイトは簡単・手軽が命。使えるからといってごちゃごちゃとコムズカシイ技に走るより，基本のタグだけをしっかり押さえるほうが見る人に優しく，また作る人にとってもラクなのは間違いありません。



《第2図》これまたポピュラーな構成。タグは表示画面では見えなくなる。タグの中身を書き間違うと，間違い電話になるので注意が必要だ！

醍醐味！ iモードでしか使えないタグ

やはりiモードはただモノではなかった！　iモードには，パソコンよりも機動性を重視した携帯端末ならではのタグがあります。

筆頭は，“”と“”すなわち電話をかけるタグ。クリック一発でiモードが自動的に電話をかけてくれるという，超便利機能です。なんといってもiモードは電話なのですから，電話としての機能を格段に便利にしてくれるこのタグを使わない手はないでしょう。

じつはこのタグが有名になったきっかけには，これを使った悪質ないたずらが発生してそれが大々的に報道された，というツライいきさつがありました。現行の機種での画面上の表示は次ページへのリンクも電話機能も等しく，下線表示にされてしまうので，不親切な書きかたをされると本当に紛らわしいことがあるのです。それも事件を大きくしてしまった理由の一つかもしれません。

とりあえずはタグの間に挟む文字列を「こちら」とか「電話」だけにしないで，実際に番号を書くなどして，はっきりアピールするべきでしょう。そういう工夫があるということが，より良質なホームページであることの目安とも言えます。

ところで，電話ができるならメールもクリック一発でできないかなと考えたらスルドイです。メールタグもあるんです。これはiモード専用ではなく一般サイトでも使われるタグですが，“”と“”で，書きかたは電話タグといっしょ。こちらもタグの間にメールアドレスの文字列などを挟んでおくとよいでしょう（**リスト2，第2図**）。

タグ＝見栄えと心得よ

ホームページはどれもこれらの基本的

裏技コラム　Q.iモードで一般のパソコン向けサイトを見ることはできるの？

答えは「できるものはできるけど，できないものはできない」。

iモード専用サイトと一般のパソコン向けサイトは，ページを作っている言語が違うということは本文でも触れましたが，この違いを克服することは一部では可能です。

『通勤ブラウザ（http://www.sjk.co.jp/）』というページでは，一般のホームページを携帯端末から見ることができるサービスを提供しています。そしてこのアドレスから検索ページを利用すれば，携帯端末でもパソコンを使った一般のサイトに自由にアクセスすることができるようになるのです。

それでもすべて見ることができるわけではありません。iモードの表示画面の限界を超えてしまう大きな画像は見られませんし，容量の大きすぎる動画も同様です。また，「フレーム」という画面分割の手法も表示画面の小さな携帯端末ではそもそも表示が不可能なので除外です。

でもそれ以外の部分，つまり文字は見ることができるので「読むこと」＝文字情報にかけては問題ないと言えます。また，文字を使ってのさまざまな作業，たとえば検索，掲示板への投稿，チャットなども携帯端末から行えますので，参加することの「幅」に関してはあまり制限されることはありません。

なタグをうまく使いこなすことで長文のコンテンツをカッコよくレイアウトしています。が，ここで気をつけなければならない重要なポイントは「タグとはあくまでも画面の見栄えをよくするために使うもの」ということです。

たとえば**第1図**。タイトルの「いろいろなタグを使おう」には“<center>”タグを使っていますが，文字数が10文字だったために2文字はみ出た妙な恰好になってしまってます。

ｉモードの画面は横にスクロールすることができません。ですから現行の機種が表示できる「１行」の幅を常に意識してタグを使わなければ，こういうおマヌケ画面ができてしまいます。もちろん，画面には10文字表示のタイプも登場していますので，8文字表示で大丈夫だったものが10文字タイプで表示したらガタガタに，なんてことも起こります。

これらを避けるには，長めの文字列をセンタリング（中央揃え）する場合，あらかじめキリのいい場所に“
”を入れておくとか，ｉモードならではの機能＝半角カナを活用して文字数そのものを調整するなどの手があります（**リスト3**，**第3図**）。また，長文では段落のおわりが画面の端にぴったりきてしまうこともありますが，つぎの段落の開始が一字下がっていれば読み手が混乱することはないでしょう。

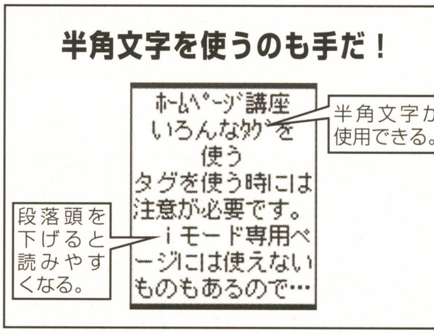
まとめ

今回紹介した，“
”，“<p>”，“<center>”，“<hr>”，そして電話タグとメールタグ。これだけでも既存のページのレイアウトのほとんどをカバーできることがお分かりいただけたでしょうか。タグは特にｉモードの画面上では拍子抜けするほど簡単に使いこなせるようになります。だからこそ，ｉモードは初心者にやさしいツールであると，ワタクシ胸を張って言い続けてきたのです。

次回は，作ったコンテンツを増やしてつなげ，複数のページを持つ立派なホームページへと仕立て上げる方法を伝授します。お楽しみに！

《リスト3》半角カナを使用し，長めのタイトルに
を使い，段落の切り替わりを一字下げたリスト

```
<html>
<head>
<title>表示テスト</title>
</head>
<body>
<center>
ﾎｰﾑﾍﾟｰｼﾞ 講座
</center>
<center>
いろんなﾀｸﾞ を<br>
使う
</center>
タグを使う時には
注意が必要です。<br>
　ｉモード専用ページには使えないものもあるので…
</body>
</html>
```



《第3図》文字の改行など気をくばると見栄えのよいものができる



今回はネット麻雀「東風荘」に挑戦！

麻雀もネットゲームもほとんど初心者の私。かなり不安だったんですが登録もあっという間にできました。

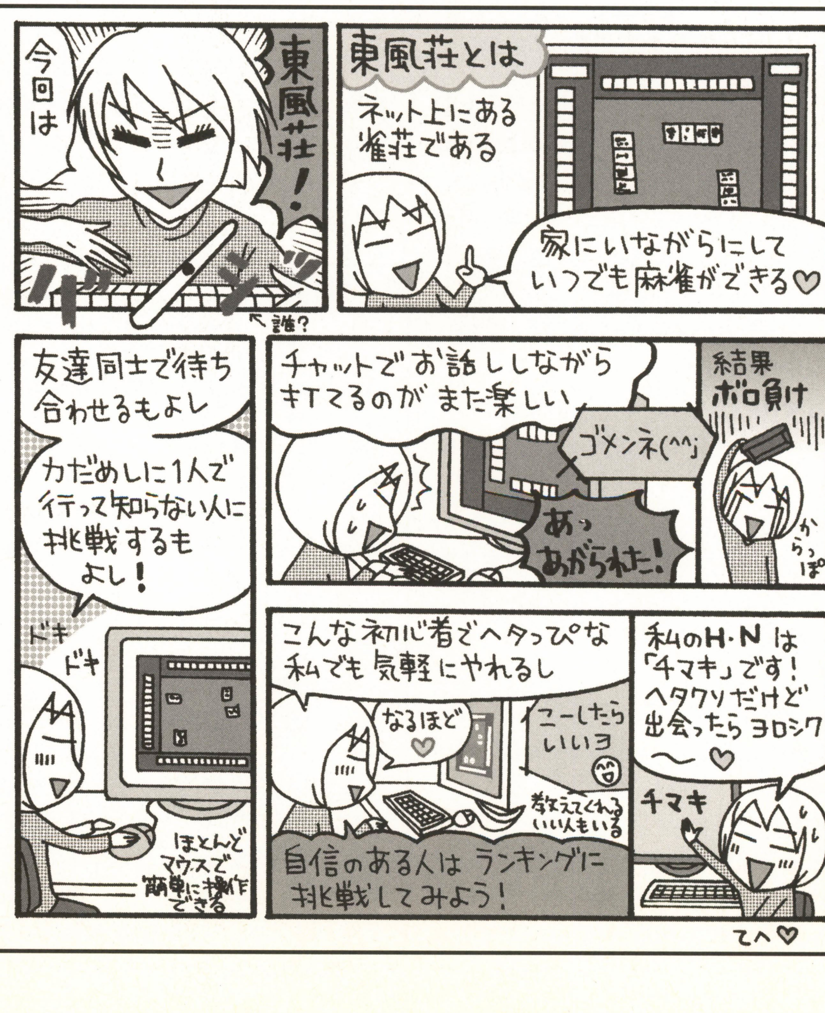
私は初心者ということで，チャット中心の初心者コースに行きましたが，ランキングに参加できる上級コースもあり，そこでは毎日熱い闘いがくりひろげられているそうです。

じつは一度本物の雀荘にも行ったことがあるのですが，ネットの雀荘もほとんど本物の雀荘で打つのと変わりなかったです。違うところと言えば実際に顔をあわせないことだけでしょうか。

相手は人間・・何が起るか分からないそのドキドキ感がまた，たまりません。

やってみたーいという方はhttp://mj.giganet.net/bitparade/にアクセスしてね！

（神奈川）



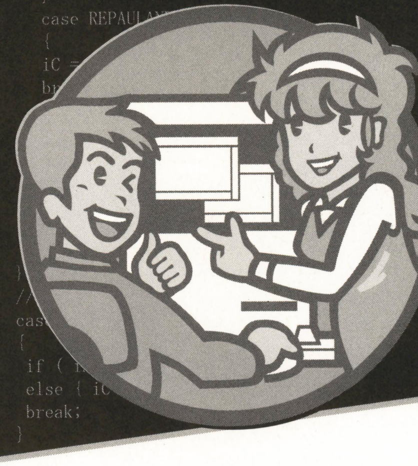
プログラマー&クリエイターへの登竜門

BM特選 投稿プログラムコーナー

- Visual Basic
- DirectX for Visual Basic
- VB Script + DirectX
- HSP
- Java
- PC-9801 N88-BASIC(86)
- FX-890P
- MUSIC

本コーナーは，さまざまな言語に対応したプログラミング学習と創作をテーマに，読者からの投稿作品で成り立っています。これから，プログラミングを勉強しようというみなさんや，プロのプログラマー／クリエイターを目指そうというみなさんにとって役立つ情報が満載です。手近にあるパソコンに入力して，お楽しみください。

それぞれの記事にはプログラムを入力／実行するときの注意点やプログラムの内容，改造方法などが紹介されていますので，よく読んでください。また，“CHECKER FLAG”というコラムがあり，デバッグ博士ことDr.Dがそれぞれのプログラムに対し批評を加えています。口が悪い人なので，かなりキビシイことも言っていますが，自分でプログラムを作ったり，これから投稿しようとしている人にとっては，とても参考になるでしょう。



プログラム投稿者へのお願い

自分の作ったプログラムを投稿しようという人はつぎのきまりを守って応募してください。

①プログラム（作品）ファイル ②プログラムの説明，住所・氏名・年齢・職業（学校名）・電話番号（連絡のつく時間も）を書いた原稿のテキスト・ファイルをフロッピーディスクにセーブし（プリンタで打ち出した原稿も同封のこと），お送りください。

プログラムの説明には，①内容 ②使いかた・遊びかた ③操作方法 ④プログラムのしくみ（変数表やフローチャートも）⑤プログラムの具体的入力方法，注意点 ⑥改造のアドバイス，作ったとき苦労した点 ⑦使用機種名 ⑧言語（バージョン，モードなども）などを書きます。

注1）プログラムは必ずチェックし，ミスを防ぐために同じものを2～3入れてもらえば安心です。

注2）プログラムの説明は，テキスト・ファイルをフロッピーディスクにセーブし，念のためプリンタで打ち出したものも同封することが基本ですが，手書き原稿でも受け付けます。その際，できるだけていねいに，マス目のある用紙を使用してください。

注3）フロッピーディスクのラベル，封筒の裏面にもプログラム名（ファイル名も），原稿ファイル名，住所，氏名，電話番号を必ず記入してください。

内容はオリジナルのもので，未発表のものに限ります。もし，参考にした記事や本がありましたら，書名，号数，記事名，筆者名，記事のタイトルなども明記しておいてください。

お送りいただいたディスクはお返しできませんので，あらかじめバックアップをとっておいてください。

掲載されたプログラムに対しては，規定の原稿料をお支払いし，掲載誌をお送りいたします。　**編集部**

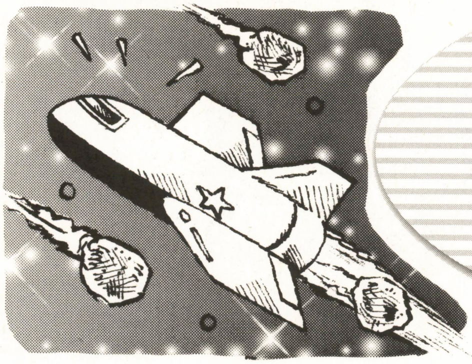
★本誌とじ込みの愛読者カードについている応募カードも忘れずに同封してください。

あて先：〒141-8715 東京都品川区東五反田1-11-15 電波新聞社マイコンBASICマガジン「特選プログラム・コーナー」係

★ここに掲載されているプログラムは，ベーマガ・ホームページからもダウンロードできます。また，ホームページからの投稿も受け付けています。

http://www.basicmagazine.net/

Windows(Visual Basic V6.0)用



接近する隕石をひたすらよけろ

YOKE ZONE（ヨケ ゾーン）

吉岡　敬浩（12歳）

内容&ストーリー

21XX年，宇宙旅行をしていたオヤジたちがいた。はじめは旅行は順調だった。しかし，ある惑星をすぎたあたりのことだった……

オヤジ01:「ん？　なんだ？　隕石がこちらへ来ているぞ!!」

オヤジ02:「これがウワサの，突入したら２度と出ることができないデッドゾーンか」

《第１表》変数表

Player(x.y)	自機の座標
R,L,U,D,X,YiSpk	キー入力関係
teki(8)	敵の存在フラグ
switch	敵の出現速度関係
x2(8),y2(8),x22(8),y22(8),play1,play2	座標関係
life	自機ライフ
score	スコア
title	タイトル時のフラグ
sx(8),sy(8)	背景星座標
sflg(8)	星の存在フラグ
tswitch	星の出現速度関係

オヤジ01:「よし，できるかぎり生き延びよう!!」

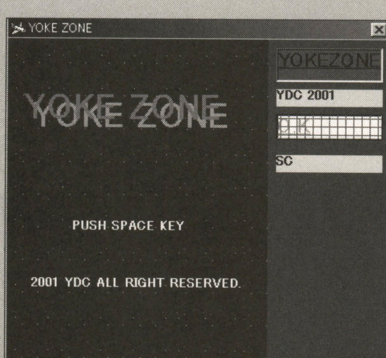
というわけで一見シューティング・ゲームに見えるけど，よけゲーです。

操作方法

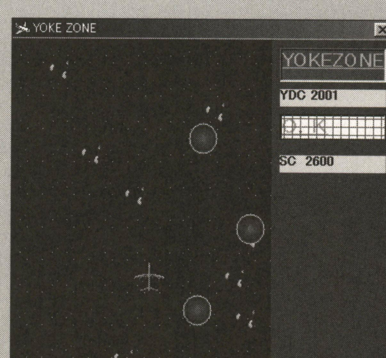
タイトル画面でスペース・キーを押すとゲーム・スタートです。自機はカーソル・キーで前後左右に動きます。

改造のアドバイス

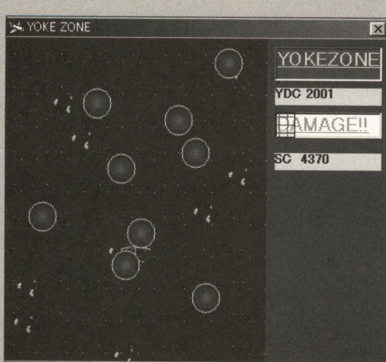
マウスやジョイスティック(パッド)でも操作できるようにする。効果音をもっとダイナミックにする。



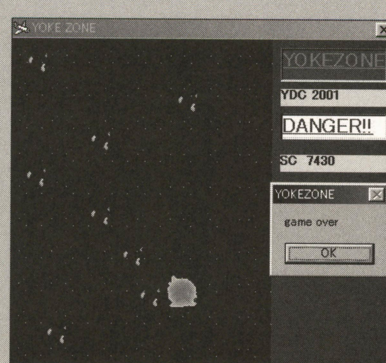
《写真１》小学生が作ったVBのゲームプログラム



《写真２》画面上から現れる隕石をよけるのだ



《写真３》隕石の数は先へ進むに比例して多くなる



《写真４》ハイスコアを目指してプレイしよう

CHECKER FLAG

編：3月号の『落とし合い』に続いて，この作品も小学生プログラマーからの投稿作品。こういった年齢層からの投稿が続くと，これから先のベーマガへの投稿が今から楽しみになってきますね。

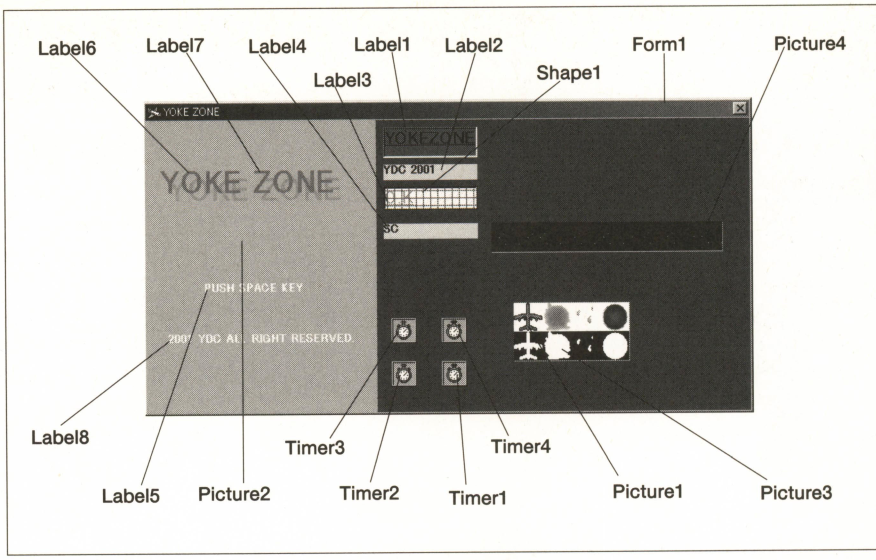
つぐ美：内容もゲームとして基本的なシステムはキチンと組み込まれているわ。某ドリキャスRPGソフトのビジュアルメモリのミニゲームでも，こういった障害避けのゲームがありましたしね。

Dr.D：リストの組みかたもコントロールのイベント処理のみをfrmファイルに記述し，ゲームシステムはモジュールファイルに記述するという，VBゲームプログラムの王道的な書きかたをしており，キレイなリストと言える。じゃが説明にあるとおり，このモジュールにあるプロシージャの記述はもう少し整理できそうじゃな。この部分は次回作への課題にしてはどうかな？

《OVER FLOW》4月号からの誌面刷新に関して，読者のみなさんから，多くの意見，感想，提案をいただきました。ベーマガは当面，プログラミングを中心にして広がっている事象に関して，その楽しさを多くの方々に伝えていきたいと考えます。よろしくお願いします。

《第２表》フローチャート

変数などの宣言
↓
タイトル画面の処理 （下からメインループ） 星を作成
↓
背景を描画
↓
星を描画
↓
プレーヤーのキー入力の処理
↓
飛行機の描画
↓
敵の作成
↓
敵の移動
↓
敵の描画
↓
画面更新
↓
（星を作成に戻る）



フォームの配置図

最後に

初めてVBで完成することができたゲームです。だから，かなりリストがめちゃめちゃだったりしてます。ちなみに僕のハイスコアは15,321点です。

これからもいろいろなゲームをつくりたいです。感想などはtakahiro@arsystem.net,ホームページはhttp://www.arsystem.net/~takahiroです。

★参考にしたもの

・ＶＢの入門書

・インターネット

・ベーマガ

《第３表》プロパティ表

フォームの設定	
Name	Form1
AutoRedraw	-1 'True
BackColor	&H00404040&
BorderStyle	3 '固定ダイアログ
Caption	YOKE ZONE
Font	ＭＳ Ｐゴシック
Icon	(アイコン)
Height	5175
Left	0
KeyPreview	-1 'True
LinkTopic	Form1
MaxButton	0 'False
MinButton	0 'False
ScaleHeight	320
ScaleMode	3 'ピクセル
ScaleWidth	671
ShowInTaskbar	0 'False
StartUpPosition	3 'Windows の既定値
Top	0
Width	10155
タイマーの設定	
Name	Timer1
Interval	1000
Left	328
Top	264
Name	Timer2
Interval	1000
Left	272
Top	264
Name	Timer3
Interval	10
Left	272
Top	216
Name	Timer4
Interval	1000
Left	328
Top	216
ピクチャボックスの設定	
Name	Picture1
AutoRedraw	-1 'True
BorderStyle	0 'なし
Font	ＭＳ Ｐゴシック
Height	33
Left	408
Picture	(ビットマップ)
ScaleHeight	33
ScaleMode	3 'ピクセル
ScaleWidth	129
TabIndex	0
Top	200
Width	129
Name	Picture2
AutoRedraw	-1 'True
BorderStyle	0 'なし
Font	ＭＳ Ｐゴシック
Height	320
Left	0
ScaleHeight	320
ScaleMode	3 'ピクセル
ScaleWidth	256
TabIndex	1
Top	0
Width	256
Name	Picture3
AutoRedraw	-1 'True
BorderStyle	0 'なし
Font	ＭＳ Ｐゴシック
Height	33
Left	408
Picture	(ビットマップ)
ScaleHeight	33
ScaleMode	3 'ピクセル
ScaleWidth	129
TabIndex	2
Top	232
Width	129
Name	Picture4
AutoRedraw	-1 'True
BorderStyle	0 'なし
Font	ＭＳ Ｐゴシック
Height	33
Left	384
Picture	(ビットマップ)
ScaleHeight	495
ScaleWidth	3855
TabIndex	3
Top	112
Width	257
ラベルの設定	
Name	Label1
BackColor	&H00404080&
BorderStyle	1 '実線
Caption	YOKEZONE
Font	ＭＳ Ｐゴシック
Height	33
Left	264
TabIndex	4
Top	8
Width	105
Name	Label2
BackColor	&H0080FF80&
Caption	YDC 2001
Font	MS UI Gothic
Height	17
Left	264
TabIndex	5
Top	48
Width	105
Name	Label3
BackColor	&H00C0FFFF&
BorderStyle	1 '実線
Caption	O.K
Font	ＭＳ Ｐゴシック
ForeColor	&H0000C000&
Height	25
Left	264
TabIndex	6
Top	72
Width	105
Name	Label4
BackColor	&H0080FF80&
Caption	SC
Height	17
Left	264
TabIndex	7
Top	112
Width	105
Name	Label5
BackStyle	0 '透明
Caption	PUSH SPACE KEY
ForeColor	&H0000FFFF&
Height	17
Left	64
TabIndex	8
Top	176
Width	153
Name	Label6
BackStyle	0 '透明
Caption	YOKE ZONE
Font	ＭＳ Ｐゴシック
ForeColor	&H008080FF&
Height	33
Left	24
TabIndex	9
Top	56
Width	209
Name	Label7
BackStyle	0 '透明
Caption	YOKE ZONE
Font	ＭＳ Ｐゴシック
ForeColor	&H000000FF&
Height	33
Left	16
TabIndex	10
Top	48
Width	209
Name	Label8
BackStyleq	0 '透明
Caption	2001 YDC ALL RIGHT RESERVED.
ForeColor	&H0000FFFF&
Height	17
Left	24
TabIndex	11
Top	232
Width	233
シェイプの設定	
Name	Shape1
FillColor	&H00C000C0&
FillStyle	6 'クロス
Height	25
Left	264
Top	72
Width	105

★PSを知らないのに，SONYのAVマルチ入力端子付きテレビを購入するお父さんって……。でもAVマルチ用のコードがなかなか売っていなくて，結局，普通のきれいなテレビになっています。（愛知県刈谷市・**りんす26歳**）……【**影：**きれいが何より。私のテレビは…。】

《リスト１》frmファイルのプログラム・リスト

```
Option Explicit
Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
Select Case KeyCode
    Case vbKeyRight
        R = YEPUSH  '右
    Case vbKeyLeft
        L = YEPUSH  '左
    Case vbKeyUp
        U = YEPUSH  '上
    Case vbKeyDown
        D = YEPUSH  '下
    Case vbKeySpace
        iSpk = YEPUSH  'スペースキー
End Select
End Sub
Private Sub Form_KeyUp(KeyCode As Integer, Shift As Integer)
Select Case KeyCode
    Case vbKeyRight '右
        R = NOPUSH
    Case vbKeyLeft  '左
        L = NOPUSH
    Case vbKeyUp    '上
        U = NOPUSH
    Case vbKeyDown
        D = NOPUSH  '下
    Case vbKeySpace
        iSpk = NOPUSH 'スペースキー
End Select
End Sub
Private Sub Form_QueryUnload(Cancel As Integer, UnloadMode As Integer)
    End
End Sub
Private Sub Timer1_Timer()
    switch = True: Label1.ForeColor = QBColor(Int(Rnd * 15))
End Sub
Private Sub Timer2_Timer()
    If Timer1.Interval = 100 Then Timer2.Enabled = False
    Timer1.Interval = Timer1.Interval - 50
End Sub
Private Sub Timer3_Timer()
    score = score + 10: Form1.Label4.Caption = "SC  " & score
End Sub
Private Sub Timer4_Timer()
    tswitch = True
End Sub
```

《リスト２》モジュールファイルのプログラム・リスト

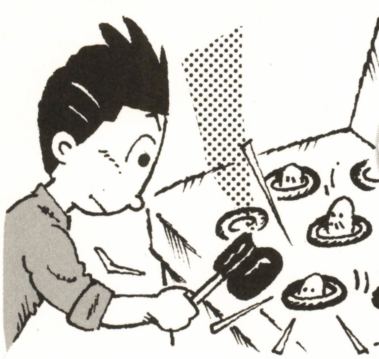
```
Option Explicit
Public Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
Public Declare Function BitBlt Lib "gdi32" (ByVal hDestDC As Long, ByVal X As Long, ByVal Y
 As Long, ByVal nWidth As Long, ByVal nHeight As Long, ByVal hSrcDC As Long, ByVal xSrc As
Long, ByVal ySrc As Long, ByVal dwRop As Long) As Long
Public hakox As Integer
Public Type Player
    X As Integer
    Y As Integer
End Type
Public Player As Player
Public R As Integer
Public L As Integer
Public U As Integer
Public D As Integer
Public X As Integer
Public Y As Integer
Public Const NOPUSH = 0
Public Const YEPUSH = 1
Public teki(8) As Boolean
Public switch As Boolean
Public x2(8) As Integer
Public y2(8) As Integer
Public x22(8) As Integer
Public y22(8) As Integer
Public play1 As Integer
Public play2 As Integer
Public life As Integer
Public score As Long
Public title As Boolean
Public iSpk As Integer
Public sx(8) As Integer
Public sy(8) As Integer
Public sflg(8) As Boolean
Public tswitch As Boolean
Public Function HitORMis(x1, y1, xSize1, ySize1, x3, y3, xSize2, ySize2) As Boolean
    If x1+xSize1>=x3 And x1<=x3+xSize2 And y1<=y3+ySize2 And y1+ySize1>=y3 Then
        HitORMis = True
    Else
        HitORMis = False
    End If
End Function
Sub Main()
    Form1.Timer3.Enabled = False: Form1.Timer2.Enabled = False: Form1.Timer1.Enabled=False
    Form1.Show: Player.X=120: Player.Y=250: title=True: Form1.Width=5715
    drawback
Do 'タイトル画面
DoEvents
    spacec
    If title = False Then Exit Do
    Sleep 30
Loop
    Form1.Label5.Visible = False: Form1.Label6.Visible = False
    Form1.Label7.Visible = False: Form1.Label8.Visible = False
    Form1.Timer3.Enabled = True: Form1.Timer2.Enabled = True: Form1.Timer1.Enabled = True
Do 'メインループ
DoEvents
    cstar
    drawback
    drawstar
    MovePlayer
    DrawPlayer
    tcheck
    moveteki
    drawteki
    Form1.Picture2.Refresh
    Sleep 20 'ウエイト
Loop
End Sub
Sub MovePlayer()
    If U = YEPUSH Then
        Player.Y = Player.Y - 8
        If Player.Y < 0 Then
            Player.Y = 0
        End If
    End If
    If D = YEPUSH Then
        Player.Y = Player.Y + 8
        If Player.Y > 288 Then
            Player.Y = 288
        End If
    End If
    If R = YEPUSH Then
        Player.X = Player.X + 8
        If Player.X > 224 Then
            Player.X = 224
        End If
    End If
    If L = YEPUSH Then
        Player.X = Player.X - 8
        If Player.X < 0 Then
            Player.X = 0
        End If
    End If
End Sub
Sub DrawPlayer()
    BitBlt Form1.Picture2.hDC, Player.X, Player.Y, 32, 32, Form1.Picture3.hDC, 0, 0, vbSrcPaint
    BitBlt Form1.Picture2.hDC, Player.X, Player.Y, 32, 32, Form1.Picture1.hDC, 0, 0, vbSrcAnd
End Sub
Sub drawback()
    For hakox = 0 To 10
        BitBlt Form1.Picture2.hDC, 0, hakox*32, 256, 32, Form1.Picture4.hDC, 0, 0, vbSrcCopy
    Next hakox
End Sub
Sub tcheck()
    Dim t
    For t = 0 To 8
        If y2(t) = 320 Then
            teki(t) = False
            y2(t) = 0
        End If
    Next t
    If switch = False Then Exit Sub
    switch = False: Randomize
    For t = 0 To 8
        If teki(t) = False Then
            teki(t) = True: GoTo 1
        End If
    Next t
    Exit Sub
1   '仕方ない
        x2(t) = Int(Rnd * 224)
End Sub
Sub moveteki()
    Dim t
    For t = 0 To 8
        If teki(t) = True Then y2(t) = y2(t) + 8
    Next t
End Sub
Sub drawteki()
    life = Form1.Shape1.Width * 100: hakox = Form1.Shape1.Width
    If life <= 2000 Then
        Form1.Label3.Caption = "DANGER!!": Form1.Label3.ForeColor = QBColor(4)
    Else
        Form1.Label3.Caption = "O. K"
    End If
    If life <= 100 Or hakox <= 2 Then
        gameover
    End If
    Dim t
    For t = 0 To 8
        If teki(t) = True Then
            x22(t) = 32: y22(t) = 32: play1 = 24: play2 = 24
            If HitORMis(Player.X+3, Player.Y+3, play1, play2, x2(t), y2(t), x22(t), y22(t)) Then
                Beep
                Form1.Label3.Caption = "DAMAGE!!": life = life - 10
                Form1.Shape1.Width = Form1.Shape1.Width - 1.05
            End If
            BitBlt Form1.Picture2.hDC, x2(t), y2(t), 32, 32, Form1.Picture3.hDC, 96, 0, vbSrcPaint
            BitBlt Form1.Picture2.hDC, x2(t), y2(t), 32, 32, Form1.Picture1.hDC, 96, 0, vbSrcAnd
        End If
    Next
End Sub
Sub gameover()
    Form1.Timer3.Enabled = False
    Form1.Timer2.Enabled = False
    Form1.Timer1.Enabled = False
    BitBlt Form1.Picture2.hDC, Player.X, Player.Y, 32, 32, Form1.Picture4.hDC, 32, 0, vbSrcCopy
    BitBlt Form1.Picture2.hDC, Player.X, Player.Y, 32, 32, Form1.Picture3.hDC, 32, 0, vbSrcPaint
    BitBlt Form1.Picture2.hDC, Player.X, Player.Y, 32, 32, Form1.Picture1.hDC, 32, 0, vbSrcAnd
    Form1.Picture2.Refresh
    Sleep 300
    MsgBox "game over"
    End
End Sub
Sub spacec()
    If iSpk = YEPUSH Then
        Beep
        title = False
    End If
End Sub
Sub drawstar()
    Dim t
    For t = 0 To 8
        If sflg(t) = True Then
            sy(t) = sy(t) - 1
        End If
    Next t
    t = 0
    For t = 0 To 8
      If sflg(t) = True Then
          BitBlt Form1.Picture2.hDC, sx(t), sy(t), 32, 32, Form1.Picture3.hDC, 64, 0, vbSrcPaint
          BitBlt Form1.Picture2.hDC, sx(t), sy(t), 32, 32, Form1.Picture1.hDC, 64, 0, vbSrcAnd
      End If
    Next
End Sub
Sub cstar()
    Dim t
    For t = 0 To 8
        If sy(t) = 0 Then
            sflg(t) = False: sy(t) = 320
        End If
    Next t
    If tswitch = False Then Exit Sub
    tswitch = False: Randomize
    For t = 0 To 8
        If sflg(t) = False Then
            sflg(t) = True: GoTo 1
        End If
    Next t
    Exit Sub
1   '仕方ねー
        sx(t) = Int(Rnd * 224)
End Sub
```


★「砦の攻防」はよく移植されますけど，いちばん最初のバージョンは，どの機種で，いつ頃掲載されたのでしょうか？（群馬県勢多郡・Noo16歳）……**【影：するどい感想ですね。ちょっと資料室で調べものしてきまーす。つぐ美：また昼寝しに行くわけね。】**

Windows(Visual Basic V6.0)用

短いリストで作られたゲーム

VB(ブイビー)もぐら



バビブベボーちゃん(19歳)

内容

もぐらたたきです。コマンドボタンにもぐらに見立てた“@”が表示されたら，そのボタンをクリックしてください。“@”が消えてスコアが+1されます。

逆に何も表示されていないボタンをクリックするとスコアが-1されます。

とくにクリア条件などを設けなかったため，暇つぶし感覚で遊んでください。

遊びかた

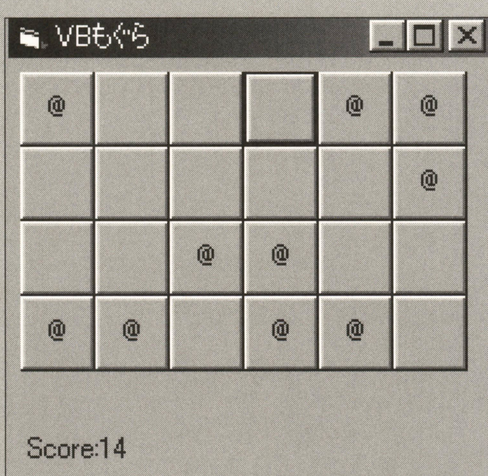
プログラム入力後，VBの実行ボタンを押す，もしくはダウンロードファイルにあるexeファイルを実行すると，ゲームが始まります。マウスのみを使用してゲームをプレイしていきます。

操作方法は左クリックでコマンドボタンを押す(=もぐらを叩く)のみですので，直感的に操作できます。

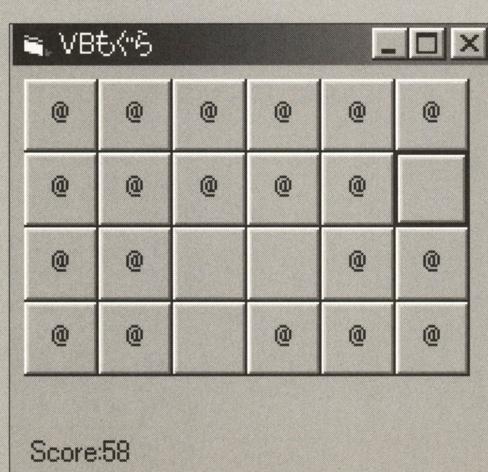
入力方法

VBを起動後，標準EXEを選択してプログラムを作成できる状態にします。まず，はじめにフォームに“コマンドボタン”，“ラベル”，“タイマー”を配置します。

そして，掲載されているプロパティ表どおりに各コントロールのプロパティを設定していきます。具体的には，フォームは“Caption”，“Height”，“Width”を，コマンドボタンは“Caption”，“Height”，“Index”，“Left”，“Top”，“Width”を，ラベルは“Height”，“Left”，“Top”，“Width”を，タイマーは“Interval”を設定すれば，あとは配置したときに設定されているプロパティどおりです。



《写真１》コマンドボタンで作られたもぐらたたき



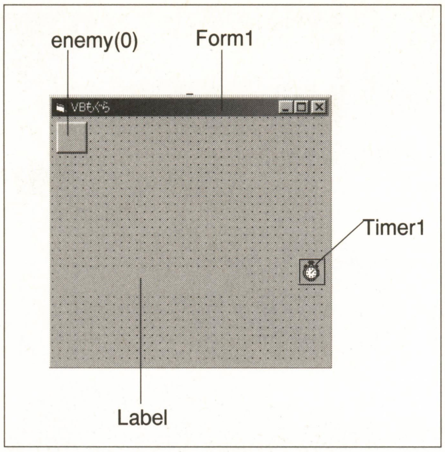
《写真２》単純な内容だが，やり始めると意外とハマルかも？

CHECKER FLAG

つぐ美：説明文にもあるとおり，ゲーム内容は単純ですが，リストがかなり短いわね。コマンドボタンのクリックという，VBの基本的な動作をゲームに取り入れているのが，リストの短さにつながっているのかしら？

Dr.D：ゲーム自体の完成度が高い作品もよいが，やはりベーマガではこういった短い作品にも注目していきたいのぉ。

★ウワー，毎月の小遣いから切手を買うのも大変です。助けて〜。（埼玉県朝霞市・Abion14歳）……【影：毎月切手を買ってるのかな？お母さんにお願いして買い置きしておこう。ちなみにAbionくんは62円切手を貼ってくれてるけど，早く届くわけじゃないよ。】



フォームの配置図

《第１表》プロパティ表

フォームの設定	
Name	Form1
Caption	VBもぐら
Heigh	3150
Left	0
LinkTopic	Form1
ScaleHeight	2745
ScaleWidth	3105
StartUpPosition	3 'Windows の既定値
Top	0
Width	3225
コマンドボタンの設定	
Name	enemy
Height	480
Index	0
Left	75
TabIndex	0
Top	75
Width	480
ラベルの設定	
Name	Label1
Height	300
Left	120
TabIndex	1
Top	2400
Width	3000
タイマーの設定	
Name	Timer1
Interval	35
Left	3840
Top	2160

その後，frmファイルのコードを表示して，リストを入力していきます。リストの入力には，各コントロールをクリックして，そのコントロールに対応したリストの部分を入力していく方法もあります(たとえばTimerだったらTimerコントロールをクリックした後，“Private Sub Timer1_Timer( )〜End Sub”の部分のリストを入力すればOK)。

後 記

VBで簡単なゲームを作成してみたいと思って作りました。ゲームアイデアはありきたりですが，その分リストはかなり短くしました(この短さはチョット自信アリ)。VB初心者プログラマーの参考になったらウレシイですね。

『VBもぐら』のプログラム・リスト

```
Const EnmSet = 100
Dim enemynum(23) As Integer
Dim score As Integer

Private Sub enemy_Click(Index As Integer)
 Dim i, j As Integer

 If enemynum(Index) = 1 Then
    enemy(Index).Caption = ""
    score = score + 1
    enemynum(Index) = 0
 Else
    score = score - 1
    If score < 0 Then
        score = 0
    End If
 End If
End Sub

Private Sub Form_Load()
    Dim i As Integer

    Randomize
    score = 0
    For i = 0 To 23
        enemynum(i) = 0
        If i <> 0 Then
            Load enemy(i)
        End If
    Next

    For i = 0 To 3
        For j = 0 To 5
            enemy(i * 6 + j).Left = (j * 32 + 5) * 15
            enemy(i * 6 + j).Top = (i * 32 + 5) * 15
            enemy(i * 6 + j).Visible = True
        Next
    Next
End Sub
'もぐらセット
Private Sub Enemy_Set()
    Dim i, n As Integer

    For i = 0 To 23
        n = Int(Rnd * EnmSet)
        If n = 0 Then
            enemynum(i) = 1
        End If

        If enemynum(i) = 1 Then
            enemy(i).Caption = "@"
        Else
            enemy(i).Caption = ""
        End If
    Next
End Sub

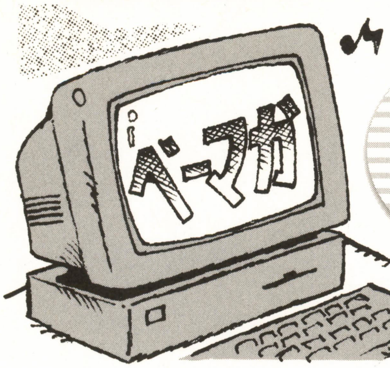
Private Sub Timer1_Timer()
    Enemy_Set
    Label1.Caption = "Score:" + CStr(score)
End Sub
```

《第２表》変数表

EnmSet	もぐらの出現確率(値が小さいと出現率アップ)
enemynum(23)	各コマンドボタンのもぐら管理用 (enemynum(0)が１だと左上のボタンにもぐらがいる)
score	スコア(得点)

★年齢がバレるのがイヤで，OFに一度も投稿していなかったが，21世紀になって『開き直る』という境地に達した。(千葉県柏市・鈴木有人28歳）……【影：開き直ったとたん，先月に引き続きの採用となりました。私も開き直って買うことにしました，馬券を。】

Windows(Visual Basic V6.0)用



実用

グラデーション文字のHTMLが作成できる

ジーツール

Gtool

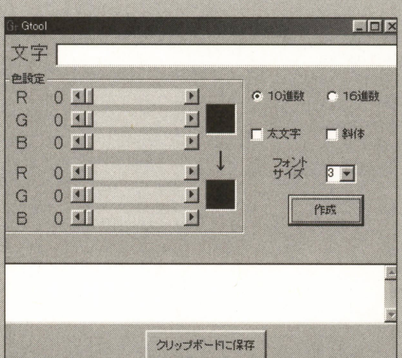
中尾　孔一（18歳）

はじめに

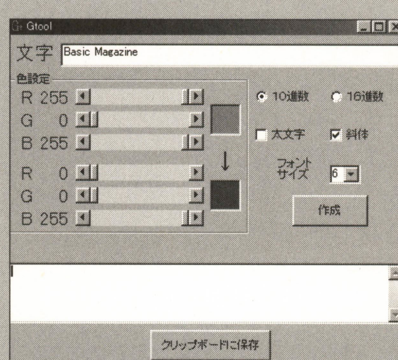
このプログラムは，HTMLでグラデーション文字を作るときに役立つツールです。最初はグラデーションを作るだけだったのですが，物足りなさを感じて今の状態になりました。初めてまともなツールを作ったなぁ。構想に1時間，制作に3日かかりました。

使いかた

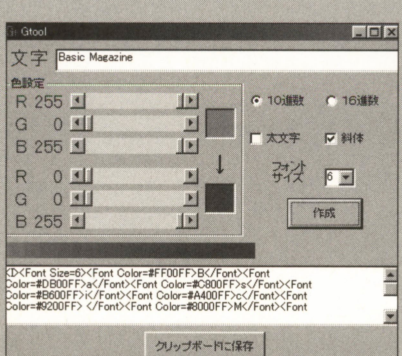
ソフトを起動したら，まず10進数か16進数を選んでください。その後，10進数ならRGBの値を，16進数なら(赤:FF00000)のように16進数で色を指定してください。Enterを押せば色が見られます。そして，一番上にHTMLで表示したい文字を入力し，必要なら太文字や斜体のチェックを入れてください。あとは，フォントサイズを指定して「作成」のボタンを押せば，ソースが表示されます。一番下のボタンを押せば，クリップボードに保存できます。



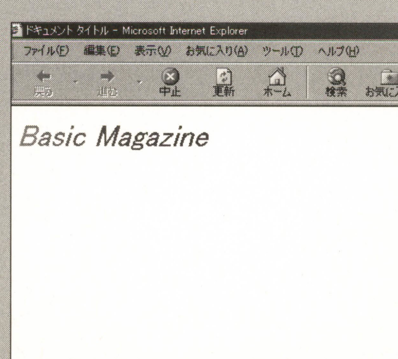
《写真１》HTMLでのグラデーション文字が作れる



《写真２》作りたい文字の情報を入力して



《写真３》作成ボタンを押せば一瞬でコードを表示



《写真４》キレイな文字列を自分のHPで使ってみよう

改造のアドバイス

(1)プレビュー機能を付ける。

(2)中間色を設定できるようにする（黒→赤→白みたいな感じに)。

両ほうとも付けたかったのですが，なにぶん受験生なので…。

苦労した点

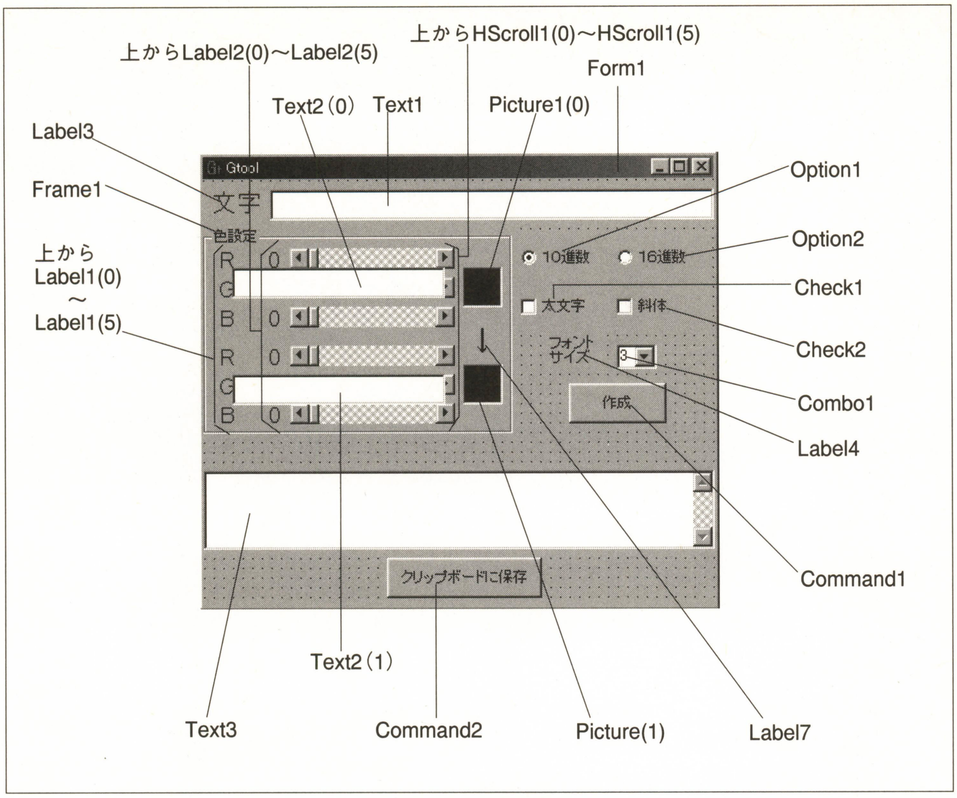
HTMLがまったくわからなかったので，自力で学習したことくらいです。あとは全部思いつきだったので，苦労したという感じがありません。

CHECKER FLAG

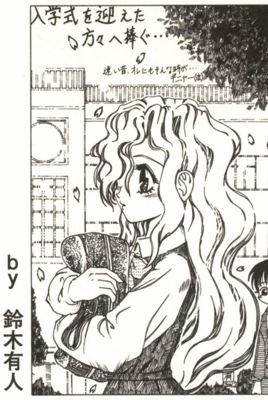
編：今までも，HTML関連のツールは何回か投稿があったのですが，実用性という面でもう一歩という作品が多かった。そういう面では，このツールは，ホームページを作るうえでの実用性が高いツールと思われます。

Dr.D：リストの面では，クリップボードの使いかたと，文字列の変数にHTMLのコードを連結していきテキストボックスへ表示する，という部分が参考になると思われるな。

つぐ美：見た感じで何となく使いかたがわかる形なので，操作性もバッチリの作品と言えるわね。



フォームの配置図



評…つい昨日のことのように思い出すわ（つぐ美）

by 鈴木有人



評…「やけくそでトーン貼ってみたり」だって（編）

by はるよん

《第１表》プロパティ表

フォームの設定	
Name	Form1
Caption	Gtool
FillStyle	0 '塗りつぶし
ForeColor	&H00000000&
Height	5670
Icon	(アイコン)
Left	0
LinkTopic	Form1
ScaleHeight	5265
ScaleWidth	6375
StartUpPosition	3 'Windows の既定値
Top	0
Width	6495
オプションボタンの設定	
Name	Option1
Caption	10進数
Height	495
Left	3960
TabIndex	6
Top	720
Value	-1 'True
Width	1095
Name	Option2
Caption	16進数
Height	495
Left	5160
TabIndex	7
Top	720
Width	1095
コマンドボタンの設定	
Name	Command1
Caption	作成
Height	495
Left	4560
TabIndex	0
Top	2520
Width	1215
Name	Command2
Caption	クリップボードに保存
Height	495
Left	2280
TabIndex	34
Top	4680
Width	1935
コンボボックスの設定	
Name	Combo1
Height	300
ItemData	(リスト)
(リスト)の内容	0,0,0,0,0,0,0
Left	5160
List	(リスト)
(リスト)の内容	1,2,3,4,5,6,7
TabIndex	32
Text	3
Top	2040
Width	495
水平スクロールバーの設定	
Name	HScroll1(0～5)
Height	255
Left	1080
Max	255
Width	2055
Name	HScroll1(0)
Index	0
TabIndex	15
Top	240
Name	HScroll1(1)
Index	1
TabIndex	14
Top	600
Name	HScroll1(2)
Index	2
TabIndex	16
Top	960
Name	HScroll1(3)
Index	3
TabIndex	17
Top	1440
Name	HScroll1(4)
Index	4
TabIndex	18
Top	1800
Name	HScroll1(5)
Index	5
TabIndex	19
Top	2160
チェックボックスの設定	
Name	Check1
Caption	太文字
Height	495
Left	3960
TabIndex	30
Top	1320
Width	1095
Name	Check2
Caption	斜体
Height	495
Left	5160
TabIndex	31
Top	1320
Width	1095
テキストボックスの設定	
Name	Text1
Height	375
Left	840
TabIndex	28
Top	120
Width	5535
Name	Text2(0～1)
Font	ＭＳ Ｐゴシック
Height	375
Left	360
ScrollBars	2 '垂直
Visible	0 'False
Width	2655
Name	Text2(0)
Index	0
TabIndex	27
Top	480
Name	Text2(1)
Index	1
TabIndex	26
Top	1800
Name	Text3
Height	975
Left	0
MultiLine	-1 'True
ScrollBars	2 '垂直
TabIndex	13
Top	3600
Width	6375
ピクチャボックスの設定	
Name	Picture1(0～1)
BackColor	&H80000007&
Height	495
Left	3240
ScaleHeight	435

★弟がSCSIボードないのに，CD-RW買っちゃいました。というわけで，プレゼントが当たりますように（3月号）。（岡山市・YasuHiro20歳）……【**影**：祈りなさい，祈れば希望がかなうのですよ。**編**：影さん，そのしゃべりかたやめてくれません？**影**：おだまり！】

ScaleWidth	435
Width	495
Name	Picture1(0)
Index	0
TabIndex	4
Top	480
Name	Picture1(1)
Index	1
TabIndex	5
Top	1680
フレームの設定	
Name	Frame1
Caption	色設定
Height	2535
Left	0
TabIndex	1
Top	600
Width	3855
ラベルの設定	
Name	Label1(0～5)
Alignment	2 '中央揃え
Font	ＭＳ Ｐゴシック
Height	255
Left	120
Width	375
Name	Label1(0)
Caption	R
Index	0
TabIndex	3
Top	240
Name	Label1(1)
Caption	G
Index	1
TabIndex	8
Top	600
Name	Label1(2)
Caption	B
Index	2

TabIndex	9
Top	960
Name	Label1(3)
Caption	R
Index	3
TabIndex	10
Top	1440
Name	Label1(4)
Caption	G
Index	4
TabIndex	11
Top	1800
Name	Label1(5)
Caption	B
Index	5
TabIndex	12
Top	2160
Name	Label2(0～5)
Alignment	1 '右揃え
Caption	0
Font	ＭＳ Ｐゴシック
Height	255
Left	480
Width	495
Name	Label2(0)
Index	0
TabIndex	20
Top	240
Name	Label2(1)
Index	1
TabIndex	21
Top	600
Name	Label2(2)
Index	2
TabIndex	22
Top	960
Name	Label2(3)
Index	3

TabIndex	23
Top	1440
Name	Label2(4)
Index	4
TabIndex	24
Top	1800
Name	Label2(5)
Index	5
TabIndex	25
Top	2160
Name	Label3
Alignment	2 '中央揃え
Caption	文字
Font	ＭＳ Ｐゴシック
Height	375
Left	0
TabIndex	29
Top	120
Width	855
Name	Label4
Caption	フォントサイズ
Height	375
Left	4320
TabIndex	33
Top	1920
Width	615
Name	Label7
Alignment	2 '中央揃え
Caption	↓
Font	ＭＳ Ｐゴシック
Height	375
Left	3240
TabIndex	2
Top	1200
Width	495

『Gtool』のプログラム・リスト

```
Option Explicit
Dim N, Moji, Iro
Dim R1, G1, B1
Dim R As Integer, G As Integer, B As Integer
Dim Rs, Gs, Bs

Private Sub Command1_Click()
Text3.Text = "": Moji = ""
If Option1.Value = True Then '１０進数の時
 R1 = HScroll1(0).Value: R = R1
 Rs = (HScroll1(3).Value - HScroll1(0).Value) / 100
 G1 = HScroll1(1).Value: G = G1
 Gs = (HScroll1(4).Value - HScroll1(1).Value) / 100
 B1 = HScroll1(2).Value: B = B1
 Bs = (HScroll1(5).Value - HScroll1(2).Value) / 100
 Call Byouga
ElseIf Option2.Value = True Then '１６進数の時
 If Len(Text2(0).Text) = 6 And Len(Text2(1).Text) = 6 Then
  R1 = CInt("&H" & Left(Text2(0).Text, 2)): R = R1
  G1 = CInt("&H" & Mid(Text2(0).Text, 3, 2)): G = G1
  B1 = CInt("&H" & Right(Text2(0).Text, 2)): B = B1
  Rs = (CInt("&H" & Left(Text2(1).Text, 2)) - R1) / 100
  Gs = (CInt("&H" & Mid(Text2(1).Text, 3, 2)) - G1) / 100
  Bs = (CInt("&H" & Right(Text2(1).Text, 2)) - B1) / 100
  Call Byouga
 End If
End If
End Sub
'クリップボードにソースを格納
Private Sub Command2_Click()
Clipboard.Clear
Clipboard.SetText Moji
End Sub
'１０進数のサンプルカラー表示
Private Sub HScroll1_Change(Index As Integer)
Label2(Index).Caption = HScroll1(Index).Value
R1 = HScroll1(0).Value: G1 = HScroll1(1).Value: B1 = HScroll1(2).Value
R = HScroll1(3).Value: G = HScroll1(4).Value: B = HScroll1(5).Value
Picture1(0).BackColor = RGB(R1, G1, B1)
Picture1(1).BackColor = RGB(R, G, B)
End Sub
'１０進数モードへの準備
Private Sub Option1_Click()
For N = 0 To 1
 Text2(N).Visible = False
Next
For N = 0 To 5
 HScroll1(N).Visible = True
 Label1(N).Visible = True
 Label2(N).Visible = True
Next
End Sub
'１６進数モードへの準備
Private Sub Option2_Click()
For N = 0 To 1
 Text2(N).Visible = True
Next
For N = 0 To 5
 HScroll1(N).Visible = False
 Label1(N).Visible = False
 Label2(N).Visible = False
Next
End Sub

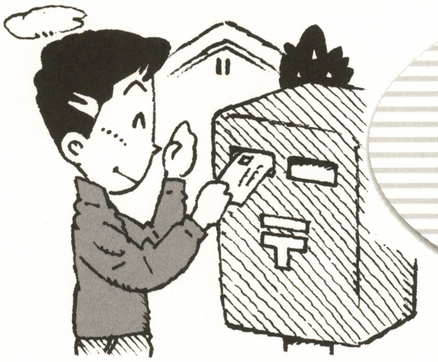
Private Sub Byouga()
'画面中央にグラデーションを表示
For N = 1 To 100
 Form1.FillColor = RGB(R, G, B)
 Form1.ForeColor = RGB(R, G, B)
 Form1.Line (N * 40, 3240)-((N + 1) * 40, 3480), , B
 R = R1 + Rs * N
 G = G1 + Gs * N
 B = B1 + Bs * N
Next
'ＨＴＭＬのソースを書く
Rs = (Rs * 100) / Len(Text1.Text)
Gs = (Gs * 100) / Len(Text1.Text)
Bs = (Bs * 100) / Len(Text1.Text)
If Check1.Value = 1 Then Moji = Moji + "<B>" '太文字
If Check2.Value = 1 Then Moji = Moji + "<I>" '斜体
Moji = Moji + "<Font Size=" & Combo1.Text & ">" 'フォントサイズ
'ここから文字のグラデーション
If Len(Hex(R1)) = 1 Then Iro = Iro + "0"
Iro = Iro + Hex(R1)
If Len(Hex(G1)) = 1 Then Iro = Iro + "0"
Iro = Iro + Hex(G1)
If Len(Hex(B1)) = 1 Then Iro = Iro + "0"
Iro = Iro + Hex(B1)
Moji = Moji + "<Font Color=#" & Iro & ">"
Moji = Moji + Left(Text1.Text, 1)
Moji = Moji + "</Font>"
Iro = ""
For N = 2 To Len(Text1.Text)
 R = R1 + Rs * N
 If Len(Hex(R)) = 1 Then Iro = Iro + "0"
 Iro = Iro + Hex(R)
 G = G1 + Gs * N:
 If Len(Hex(G)) = 1 Then Iro = Iro + "0"
 Iro = Iro + Hex(G)
 B = B1 + Bs * N
 If Len(Hex(B)) = 1 Then Iro = Iro + "0"
 Iro = Iro + Hex(B)
 Moji = Moji + "<Font Color=#" & Iro & ">"
 Moji = Moji + Mid(Text1.Text, N, 1)
 Moji = Moji + "</Font>"
 Iro = ""
Next
Moji = Moji + "</Font>"
If Check2.Value = 1 Then Moji = Moji + "</I>"
If Check1.Value = 1 Then Moji = Moji + "</B>"
Text3.Text = Moji 'ソースを表示
End Sub

Private Sub Text2_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
'１６進数のサンプルカラー表示
If KeyCode = 13 And Len(Text2(Index).Text) = 6 Then
 R = CInt("&H" & Left(Text2(Index).Text, 2))
 G = CInt("&H" & Mid(Text2(Index).Text, 3, 2))
 B = CInt("&H" & Right(Text2(Index).Text, 2))
 Picture1(Index).BackColor = RGB(R, G, B)
End If
End Sub
```

★つぐ美ちゃんは20年後20歳ということは…，今は0歳ですか？（新潟県三島郡・ゲーマー13歳）……【つぐ美：そういうことは，いちいち突っ込まなくていいのよ。自然にながしておいてね。影：そうはいかぬのですよ，ベーマガ七不思議の重点項目なわけだから。】

(超大作プログラム)　リストが誌面に掲載できない作品です。
BMホームページ(http://www.basicmagazine.net)からダウンロードしてお楽しみください。

Windows(Visual Basic V6.0)用



プログラム投稿者必須のツール

BASIC(ベイシック) Magazine(マガジン)投稿(とうこう)しようよ

谷澤　友哉（13歳）

内容

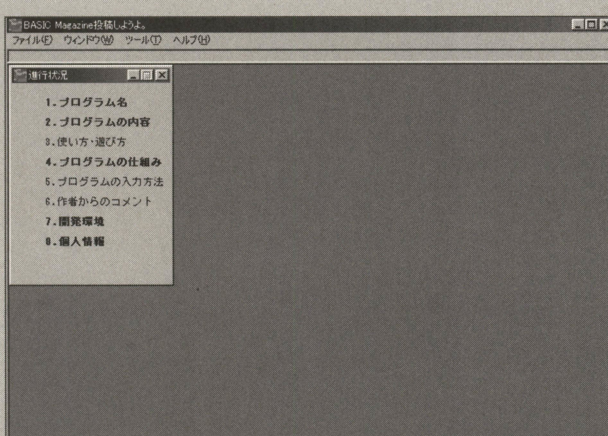
これはBASIC Magazineのプログラムを投稿する場合に書かなくてならないReadmeファイルをつくるプログラムです。ステップ方式によるデータ入力なのでReadmeファイルをレイアウトするのも簡単です。ダウンロードファイルにはヘルプセットが付属しています。ファイルの内訳は，BMRM101.Lhz:プログラム本体，HelpSet.Lhz:ヘルプセット，Source.Lhz:ソースコード一式です。

このなかでプログラムに関する詳細なReadmeは「BMRM101.Lhz」の中にあるReadmeを見てください。

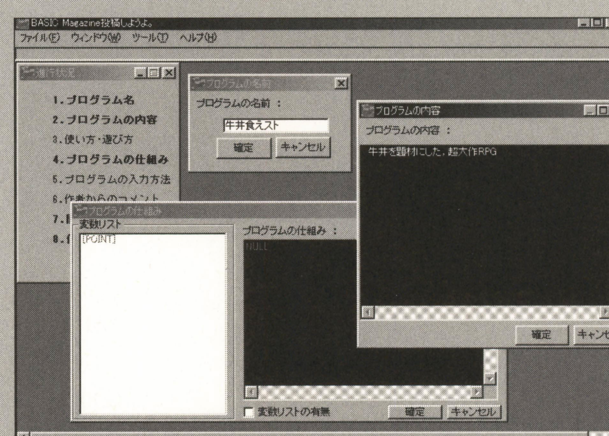
予備知識

このプログラムを使用してReadmeファイルを作成する場合に，避けれないというものが多少あります。多少ですので心配する必要はありません。簡単です。

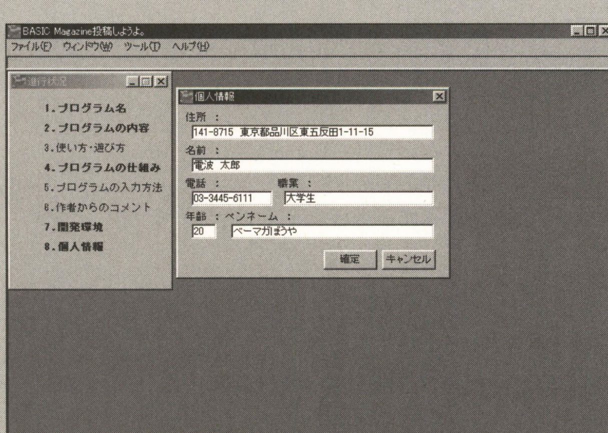
まずは，有効項目・無効項目です。こ



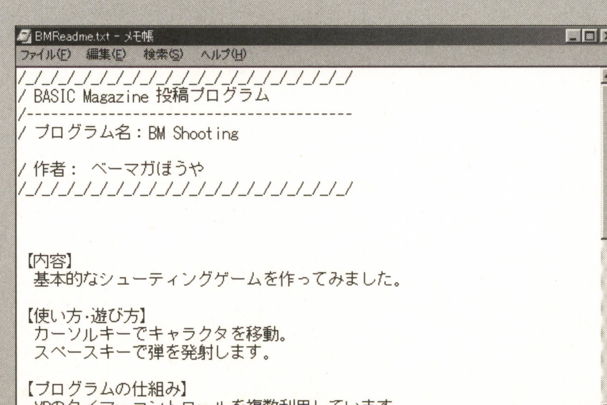
《写真１》プログラム投稿用ファイルはおまかせ



《写真２》必要な事項をどんどん入力していこう



《写真３》個人情報は別ファイルに保存可能



《写真４》こんな感じのファイルが簡単に作れちゃう

★この前テレビでソニーのロボット3体がパラパラ踊ってるのを見ました。動きがそろっていて見ていて気持ちがいい（笑）。（大阪府枚方市・松江祐輔19歳）……【つぐ美：♪ハイハイ～。ぜんぜんダメよ，こうよ。影：なぜ私が，パラパラ踊らなくちゃならないの。】

れらはReadmeファイルにその項目を落とすか，落とさないかを設定したものです。有効項目の場合は必ず記述しなければなりません。無効項目に設定できるのは有効項目以外の項目です。これらの有効項目・無効項目はステップとよばれます。つまり「項目=ステップ」なのです。参考までに言っておくと「進行状況」ウィンドウの中にあるものが項目です。項目をクリックするとそれぞれの「ステップ」ウィンドウが出てきます。「進行状況」ウィンドウの中で太字表示なのが有効項目です。

つぎに，このプログラムで重要な“NULL”について説明します。このプログラムで“NULL”とは初期値，または無変更とされます。“NULL”に設定されている場合で，有効項目でないものは無効項目とされます。有効項目の場合で“NULL”があったら，その変更はされません。

基本操作方法

まずメニューの「ファイル」-「新規作成」で情報の初期化(新規作成)を行います。

つぎに各ステップに情報を打ち込んでいきます。なお，有効項目・無効項目を問わずに，プログラムの投稿の際には書いておいたほうがいいと思います。すべてのステップを終えたら，メニューの「ツール」-「ReadMe作成」をクリックします。

以下のようなステップがあります。

・プログラムの名前
・プログラムの内容
・使いかた·遊びかた
・プログラムの仕組み
・プログラムの入力方法
・作者からのコメント
・開発環境
・個人情報

個人情報の管理

このプログラムには個人情報の管理機能があります。毎回あまり更新されない部分，つまり住所や作者名，電話番号などなど，または開発環境などの情報を管理する方法として，ファイルを使用しています。

基本的には「個人情報ファイル(*.bmm)」に保存されます。ユーザーは「ファイル」-「個人情報の保存」に保存することが可能です。

プログラムの仕組み

「BMRM101.Lhz」の中のReadmeにも記述しましたが，このプログラムのソースコードはボクの不注意でしっかりと管理されていません(変数など)。

なので変数リストを載せることができませんでした。その他のことは「BMRM101. Lhz」の中のReadmeを参照してください。

作者からのコメント

このプログラムはフリーソフトとしてやっていけるでしょうか？僕としてはフリーソフトとして公開してみたいのですが…。どうか辛口の評価のお願いします。

CHECKER FLAG

影：ベーマガへの投稿ツールというアイデアが光っている作品。超大作だけあってプログラム的な完成度も高い作品だ。

つぐ美：だけど，チョット気になる点も多いわね。その一つとして，テキスト部分に表示されるNULLの文字。文字を入力すると自動的に消えてくれたりすれば，より良いと思うけどどうかしら？また，このNULLをすべて消すと再び表示されるのもやっかいね。

編：あと，「使いかた・遊びかた」は必須項目にしてほしかったですね。この欄の記入がないために，ボツになる作品もあったりするだけに残念でした。また，記入漏れの多さで言えば郵便番号欄も欲しかったです。

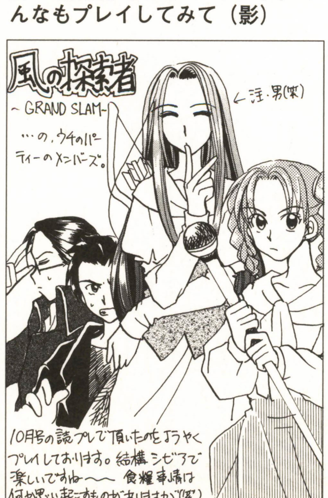
Dr.D：ベーマガ依存の実用ソフトなので，ベーマガ読者以外には，はっきり言って実用度は低い。じゃが，同じようなシステムで一般に受ける形にするには，懸賞や履歴書作成ソフトといったアイデアが考えられるかな。

評…これからはインターネットしてるときも楽しく（編）



by 鈴木真理子

評…編集部もはまってます。みんなもプレイしてみて（影）



by 都田 祥

評…イキイキとした感じを出したかったそうです（編）



by ARAI

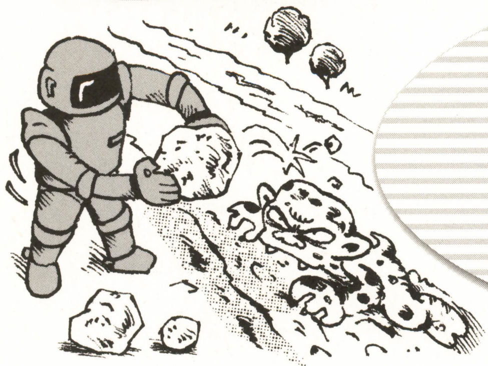
評…この制服，どこの学校がモデルかしら？（つぐ美）



by みなみとらいぱ

★3月号34ページのパソ破してしまった瞬間の影さんがかわいいと思ったのはオレだけ？（神奈川県藤沢市・たつくん16歳）……【影：わかってくれたか。こっそり，くり先生の絵を描き直しておいたんだよ。つぐ美：ウソでしょ，影さんに絵が描けるわけないもの。】

Windows(Visual Basic V5.0)用



石を武器にモンスターから森を守れ

森(もり)の守(まも)り番(ばん)

木村　雄一（19歳）

ストーリー

2XXX年，突然変異により生まれたモンスターのために，人類は絶滅の危機に瀕していた。最後に残った避難場所である断崖絶壁の森にも，モンスターの魔の手が忍び寄っていた。そこで森を守るロボットを開発し，防衛に当たらせたのであった…。

ゲームの内容

あなたは防衛ロボットを操作し，森をモンスターから守ります。崖を登ってくるモンスターを，石を落として撃退します。

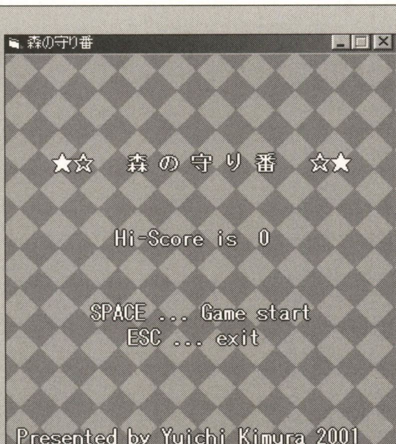
モンスターは3種類あります。

- **・白モンスター：**ただひたすら崖を登ってきます。
- **・赤モンスター：**白モンスターより高速に崖を登ってきます。
- **・青モンスター：**左右に移動しながら崖を登ってきます。

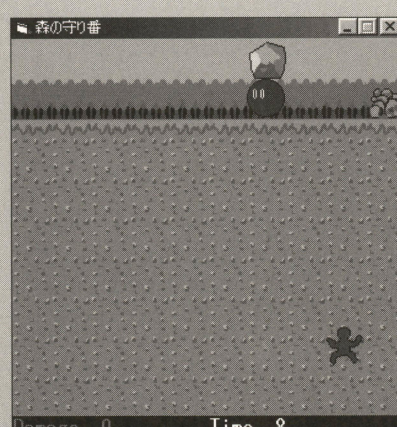
なお，時間が経過するにつれて登ってくるモンスターの数も増えていきます。モンスターが5匹登り切った時点でゲーム・オーバーとなります(左下のDamageが登り切ったモンスターの数です)。スコアは撃退したモンスターの数と守った時間で決まります。

操作方法

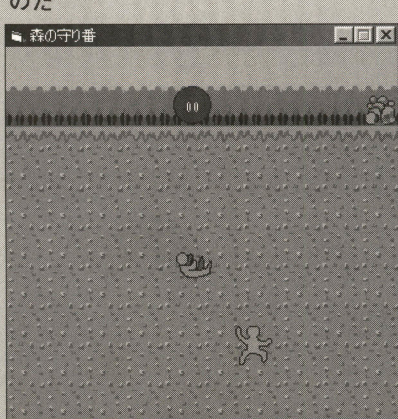
主人公はカーソル・キーの左右で動かしてください。右端にある石置き場にい



《写真１》モンスターから森を守るのだ



《写真２》下から迫ってくるモンスターに対して…



《写真３》画面右から取れる石を落とすのだ



《写真４》赤モンスターをうまく倒すのが高得点のコツ

CHECKER FLAG

影：ルールがわかりやすいため遊びやすいゲーム。敵キャラの動きも個性があるため，プレイしているとついついハマってしまうぜ。

つぐ美：細かな面では，一つの石で同じラインにいる複数の敵を落とせたり，石を落とした時点で新しい石が持てるようにするなどの処理があると，よりゲームの質が向上するんじゃないかしら？

Dr.D：frmファイルにはキー入力判定くらいしか処理がなく，内容のほとんどをモジュールで記述しており，VBではチョット変わったリストの組みかたじゃが，昔からプログラムを組んでいる人はこういう形のほうがなじみやすいじゃろう。実際，ワシもその１人だったりするがの。

★ついに僕は就職決定しました。でも，初任給までの道のりはかなり辛いですね。(福島県郡山市・近内沖尋18歳)……【一同：就職おめでとうございます。これからもがんばってください。編集部一同，陰ながら応援しています。影：給料もらったらご馳走してね。】

くと自動的に石を持ち上げます。

そしてモンスターにうまく命中するように，スペース・キーで石を落とします。石が落ちている間は，つぎの石を持てません。

入力方法

とくに難しいことはやってません。VBのTimerコントロールは不安定なので，Sub Mainを使い，Do～LoopとtimeGetTimeで擬似的なタイマー処理をしています。よって，プロジェクトのプロパティで，「スタートアップの設定」を「Sub Main」にする必要があります。

絵(char.bmp)は，一つのキャラクターを32×32で構成してください。モンスターの絵は，上から白，赤，青の順番です。

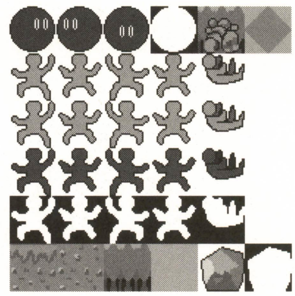
ピクチャボックスは，picMain(表示画面)とpicChar(キャラクター格納)の二つです。実行するときは，picMainの大きさにウィンドウの大きさを調節してください。

改造のアドバイス

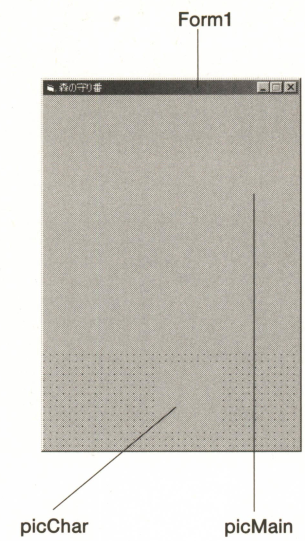
効果音をつけると面白さが増すかもしれません。ぜひ挑戦してみてください。変数enaがモンスターの出現頻度です。適当にいじって遊んでください。

最後に

VBとN88BASICを比べると，メインループの組みかたなど戸惑うことも多いですが，慣れるとVBでも楽かもしれません。絵の扱いは，やっぱりVBのほうが楽です。これからは，もっといろんなジャンルのゲームプログラミングに挑戦してみたいです。



char.bmpの画像



フォームの配置図

《第１表》プロパティ表

フォームの設定	
Name	Form1
BorderStyle	1 '固定(実線)
Caption	森の守り番
Height	5175
KeyPreview	-1 'True
Left	0
LinkTopic	Form1
MaxButton	0 'False
ScaleHeight	320
ScaleMode	3 'ピクセル
ScaleWidth	319
StartUpPosition	3 'Windows の既定値
Top	0
Width	4875
ピクチャボックスの設定	
Name	picChar
AutoRedraw	-1 'True
AutoSize	-1 'True
BorderStyle	0 'なし
Height	73
Left	144
ScaleHeight	73
ScaleMode	3 'ピクセル
ScaleWidth	73
TabIndex	1
Top	336
Visible	0 'False
Width	1095
Name	picMain
AutoRedraw	-1 'True
BorderStyle	0 'なし
Font	FixedSys
ForeColor	&H00FFFFFF&
Height	320
Left	0
ScaleHeight	320
ScaleMode	3 'ピクセル
ScaleWidth	320
TabIndex	0
Top	0
Width	320

《リスト１》frmファイルのプログラム・リスト

```
Private Sub Form_KeyDown(KeyCode As Integer, Shift As Integer)
    keymap(KeyCode) = 1
End Sub

Private Sub Form_KeyUp(KeyCode As Integer, Shift As Integer)
    keymap(KeyCode) = 0
End Sub

Private Sub Form_Unload(Cancel As Integer)
    gameflag = False
End Sub
```

《リスト２》モジュールファイルのプログラム・リスト

```
Option Explicit
Type ENEMY
    flg As Integer
    x As Integer
    y As Integer
    vx As Integer
    vy As Integer
    anim As Integer
    lv As Integer
End Type

'ゲーム進行管理フラグ
Public Const GAME_TITLE = 0
Public Const GAME_INIT = 1
Public Const GAME_MAIN = 2
Public Const GAME_END = 3
Public Const GAME_TITLE_INIT = 4
Public Const GAME_EXIT = 5

Declare Function timeGetTime Lib "winmm.dll" () As Long
Declare Function BitBlt Lib "gdi32" (ByVal hDestDC As Long, ByVal x As Long, ByVal y As Lon
g, ByVal nWidth As Long, ByVal nHeight As Long, ByVal hSrcDC As Long, ByVal xSrc As Long, B
yVal ySrc As Long, ByVal dwRop As Long) As Long

Public gameflag As Boolean
Public keymap(256) As Byte
Dim tppos As Integer
Dim nswitch As Byte, stone As Byte
Dim enm(20) As ENEMY, ena As Integer, ent As Integer   '敵データと敵の発生頻度
```

```
Dim dcount As Integer, dtime As Integer '退治したモンスター数と時間
Dim damage As Integer, tcount As Integer, ttime As Long
Dim score As Long, hiscore As Long
Dim stx As Integer, sty As Single, sta As Single    '石の座標
Dim chrx As Integer, chry As Integer, muki As Integer   'キャラクタの座標
Sub Main()
    Dim wt As Long, i As Integer

    '初期化
    Randomize
    Form1.Show
    Form1.picChar.Picture = LoadPicture("char.bmp")
    For i = 0 To 19
        enm(i).flg = False
    Next
    nswitch = GAME_TITLE_INIT
    score = 0
    hiscore = 0

    'メインループ
    gameflag = True
    Do While gameflag
        wt = timeGetTime
        DoEvents
        Select Case nswitch
            Case GAME_TITLE_INIT
                TitleInit
            Case GAME_TITLE
                GameTitle
```

★受験で東京に行ったとき，ラーメン食べたらくそまずかった。東京ではあれが普通なんですか？（大阪府河内長野市・**よっち18歳**）
……【**影：**事前に連絡くれたら，おいしいラーメン屋を案内してあげたのに！もちろん，私のオゴリでね。**つぐ美：**ヤッター！**影：**？】

```
            Case GAME_INIT
                GameInit
            Case GAME_MAIN
                GameMain
            Case GAME_END
                GameOver
            Case GAME_EXIT
                gameflag = False
        End Select
        Form1.picMain.Refresh
        While (wt + 50 > timeGetTime): DoEvents: Wend
    Loop

    End
End Sub
Sub PutTextEx(ByVal x As Integer, ByVal y As Integer, st As String)
    PutText x - 1, y, st, vbBlack
    PutText x, y - 1, st, vbBlack
    PutText x + 1, y, st, vbBlack
    PutText x, y + 1, st, vbBlack
    PutText x, y, st, vbWhite
End Sub
Sub PutText(ByVal x As Integer, ByVal y As Integer, st As String, col As Long)
    Form1.picMain.ForeColor = col
    Form1.picMain.CurrentX = x
    Form1.picMain.CurrentY = y
    Form1.picMain.Print st
End Sub
Sub GameInit()
    'ゲームの初期化
    Dim i As Integer
    For i = 0 To 19
        enm(i).flg = 0
    Next
    chrx = 0
    chry = 32
    muki = 0
    stone = 0
    ttime = 0
    tcount = 0
    ena = 90
    damage = 0
    score = 0
    ent = ena
    'スペースキーが離されるまで待つ
    If keymap(vbKeySpace) = 0 Then
        nswitch = GAME_MAIN
    End If
End Sub
Sub GameMain()
    'ゲーム本体
    Dim i As Integer, n As Integer

    'キャラクタの移動
    If keymap(vbKeyLeft) = 1 Then
        chrx = chrx - 8
        muki = 1
    End If
    If keymap(vbKeyRight) = 1 Then
        chrx = chrx + 8
        muki = 0
    End If
    If chrx < 0 Then chrx = 0
    If chrx > 288 Then chrx = 288

    '石を持ち上げる
    If chrx > 272 And stone = 0 Then
        stone = 1
    End If

    '石を持った状態
    If stone = 1 Then
        stx = chrx
        sty = 0
        '石を落とす
        If keymap(vbKeySpace) = 1 Then
            stone = 2
            sta = 0
            muki = 2
        End If
    End If

    '石が落下している
    If stone = 2 Then
        sty = sty + sta
        sta = sta + 0.4
        If sta > 10 Then sta = 10
        If sty > 320 Then
            stone = 0
        End If
    End If

    '敵を発生させる
    ent = ent - 1
    If ent = 0 Then
        For i = 0 To 19
            If enm(i).flg = 0 Then
                enm(i).flg = 1
                enm(i).x = Rnd() * 288
                enm(i).y = 320
                enm(i).lv = Int(Rnd() * 3)
                Select Case enm(i).lv
                    Case 0
                        enm(i).vx = 0
                        enm(i).vy = -1
                    Case 1
                        enm(i).vx = 0
                        enm(i).vy = -2
                    Case 2
                        enm(i).vx = 2
                        enm(i).vy = -1
                End Select
                enm(i).anim = 0
                Exit For
            End If
        Next
        ent = ena
    End If

    '敵を動かす
    For i = 0 To 19
        If enm(i).flg = 1 Then
            enm(i).y = enm(i).y + enm(i).vy
            enm(i).x = enm(i).x + enm(i).vx
            enm(i).anim = enm(i).anim + 1
            If enm(i).x < 0 Or enm(i).x > 288 Then
                enm(i).vx = enm(i).vx * -1
            End If
            If enm(i).y < 64 Then
                damage = damage + 1
                If damage = 5 Then
                    score = score * ttime
                    If score > hiscore Then
                        hiscore = score
                    End If
                    nswitch = GAME_END  'ゲームオーバー
                End If
                enm(i).flg = 0
            End If
            If enm(i).anim = 8 Then
                enm(i).anim = 0
            End If
            '石と当たっているか
            If Abs(enm(i).x-stx)<24 And Abs(enm(i).y-sty)<24 And stone=2 Then
                stone = 0
                enm(i).flg = 2
                score = score + (enm(i).lv + 1)
            End If
        ElseIf enm(i).flg = 2 Then
            enm(i).y = enm(i).y + 4
            If enm(i).y > 320 Then
                enm(i).flg = 0
            End If
        End If
    Next

    '時間の計測
    tcount = tcount + 1
    If tcount = 20 Then
        ttime = ttime + 1
        tcount = 0
        'レベルアップ
        If (ttime Mod 8) = 0 Then
            ena = ena - 5
            If ena < 30 Then ena = 30
        End If
    End If


    DrawMap

End Sub
Sub DrawMap()
    '画面描画
    Dim i As Integer, ix As Integer, iy As Integer
    Dim ret As Long, hdcmain As Long, hdcchar As Long
    hdcmain = Form1.picMain.hDC
    hdcchar = Form1.picChar.hDC

    '背景の描画
    For ix = 0 To 9
        ret = BitBlt(hdcmain, ix * 32, 0, 32, 32, hdcchar, 96, 160, vbSrcCopy)
        ret = BitBlt(hdcmain, ix * 32, 32, 32, 32, hdcchar, 64, 160, vbSrcCopy)
        ret = BitBlt(hdcmain, ix * 32, 64, 32, 32, hdcchar, 0, 160, vbSrcCopy)
        For iy = 3 To 9
            ret=BitBlt(hdcmain,ix*32,iy*32,32,32,hdcchar,32,160,vbSrcCopy)
        Next
    Next
    ret = BitBlt(hdcmain, 288, 32, 32, 32, hdcchar, 128, 0, vbSrcCopy)

    'キャラクタの描画
    ret = BitBlt(hdcmain, chrx, chry, 32, 32, hdcchar, 96, 0, vbSrcPaint)
    ret = BitBlt(hdcmain, chrx, chry, 32, 32, hdcchar, muki * 32, 0, vbSrcAnd)
    If stone <> 0 Then
        ret = BitBlt(hdcmain, stx, sty, 32, 32, hdcchar, 160, 160, vbSrcPaint)
        ret = BitBlt(hdcmain, stx, sty, 32, 32, hdcchar, 128, 160, vbSrcAnd)
    End If

    For i = 0 To 19
      If enm(i).flg = 1 Then
          ret = BitBlt(hdcmain, enm(i).x, enm(i).y, 32, 32, hdcchar, Int(enm(i).anim / 2) *
32, 128, vbSrcPaint)
          ret = BitBlt(hdcmain, enm(i).x, enm(i).y, 32, 32, hdcchar, Int(enm(i).anim / 2) *
32, (enm(i).lv + 1) * 32, vbSrcAnd)
      ElseIf enm(i).flg = 2 Then
          ret=BitBlt(hdcmain,enm(i).x,enm(i).y,32,32,hdcchar,128,128,vbSrcPaint)
          ret=BitBlt(hdcmain,enm(i).x,enm(i).y,32,32,hdcchar,128,(enm(i).lv+1)*32,vbSrcAnd)
      End If
    Next
    Form1.picMain.Line (0, 304)-(320, 320), 0, BF
    PutText 0, 304, "Damage " + Str(damage), vbRed
    PutText 160, 304, "Time " + Str(ttime), vbWhite
End Sub
Sub GameOver()
    Form1.picMain.Line (0, 96)-(320, 244), 0, BF
    PutText 120, 100, "Game Over", vbWhite
    PutText 80, 130, "Your score is " + Str(score) + "!", vbWhite
    PutText 100, 200, "Push space key", vbGreen
    If keymap(vbKeySpace) = 1 Then
        nswitch = GAME_TITLE_INIT
    End If
End Sub
Sub TitleInit()
    'タイトル画面の前準備
    tppos = 0
    'スペースキーが離された状態ならタイトルへ行く
    If keymap(vbKeySpace) = 0 Then
        nswitch = GAME_TITLE
    End If
End Sub
Sub GameTitle()
    Dim ix As Integer, iy As Integer
    Dim ret As Long, hdcmain As Long, hdcchar As Long
    hdcmain = Form1.picMain.hDC
    hdcchar = Form1.picChar.hDC
    For iy = -1 To 9
        For ix = -1 To 9
          ret=BitBlt(hdcmain,ix*32+tppos,iy*32+tppos,32,32,hdcchar,160,0,vbSrcCopy)
        Next
    Next
    PutTextEx 40, 80, "★☆  森 の 守 り 番  ☆★"
    PutTextEx 90, 140, "Hi-Score is " + Str(hiscore)

    PutTextEx 70, 200, "SPACE ... Game start"
    PutTextEx 100, 220, "ESC ... exit"

    PutTextEx 10, 300, "Presented by Yuichi Kimura 2001"

    tppos = tppos + 1
    If tppos = 32 Then tppos = 0

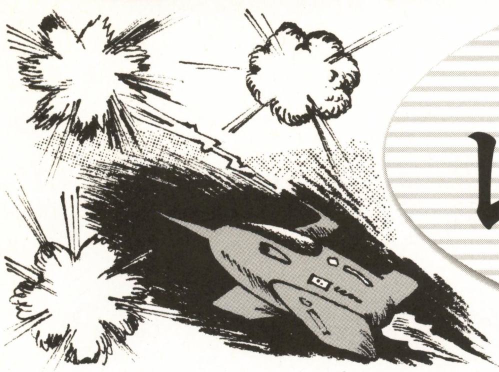
    If keymap(vbKeySpace) = 1 Then
        nswitch = GAME_INIT
    ElseIf keymap(vbKeyEscape) = 1 Then
        nswitch = GAME_EXIT
    End If

End Sub
```

★せっかく買ったグラフィックボードが，うちの機種には使えなかった…（涙）。（静岡県小笠郡・**天鳳23歳**）……【**影：**グラフィックボードのためにも，使ってもらえるパソコンをさがしてあげましょう。ちなみに私のパソコンはなんでもOKですよ。**編：**だめ！】

(超大作プログラム)　リストが誌面に掲載できない作品です。
BMホームページ(http://www.basicmagazine.net)からダウンロードしてお楽しみください。

Windows(Visual Basic V6.0+DirectX)用



定番DirectX作品の最新バージョン

レーザーシュートIII

伊藤　隆朗（15歳）

内容

2000年8月号，2001年2月号掲載の「レーザーシュート」シリーズの最新作です。今回も豪華使用，画像は「DirectDraw」，半透明にするのに「Direct3D」をちょこっと，サウンドは「DirectSoundとMusic」，キー操作に「DirectInput」を使います。

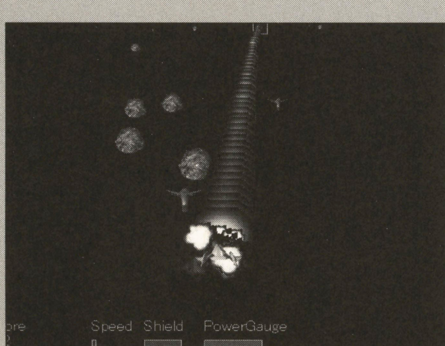
ただし，グラフィックカードの性能が悪いと半透明で表示されないようです。ゲーム自体はただ単に隕石を撃って点数を稼ぐ「II」と変わらないのですが，今回は見た目を3Dっぽくしてます。3Dサウンドを使っているので臨場感は結構なものになっていると思います。



《写真1》おなじみのDirectX作品の第3弾



《写真2》3D視点で展開されるシューティング



《写真3》必殺レーザーで敵を一気に打ち落とせ

操作方法

タイトル画面でZ，X，Cキーのどれかを押すと，いきなりゲームが始まります。カーソル・キーの左右で移動，上下でスピードを調整。Zキーで通常レーザー，Xキーで拡散レーザー，Cキーを押すと必殺レーザーが撃てます。ただし，必殺レーザーはパワーゲージを消費するのでパワーゲージがなくなれば撃てません。敵を撃てばパワーゲージはたまっていきます。

仕組み

DirectXの各機能をすべて違うモジュールにまとめてあるので，いちいち面倒なDirectXのコードを描かなくてもよいようにしています。半透明はDirectX7SDKのサンプルを切り取って作りました。ハイスコアは「HiSco.sco」というファイルに保存してあります。よかったら，matakao32@hotmail.comに送ってください。

改造点

ひたすら撃つだけでなく，内容をもっとまともにできると思います。あと，ランキングなどをつけるといいのかも…。

苦労した点

2D→3Dっぽくみせる式をたてるのに1時間半かかってしまった…。

CHECKER FLAG

編：もうVB+DirectXファンには定番(?)の『レーザーシュート』の最新版。実際，こういった続編の投稿はけっこうあるのですが，前作のマイナーチェンジにとどまってしまう作品が多く，なかなか採用レベルに至っていません。そういう意味では，この作品は画面システムを大幅に変更しているのがイイですね。

影：いっそのこと，完全な3D化にしてほしかったというのは，チョット高望みしすぎかな？この形でもなかなか迫力はあるゾ。ただ，ゲームとしては単調なので飽きやすいかもしれないな。

Dr.D：DirectXの各処理をモジュール化しているのは，キレイなリストの組みかたの見本じゃな。とくにDirectXは初期化命令がチョット面倒だったりするので，このような組みかたができるようにならないと，なかなか扱い慣れないじゃろう。

★はぁ…，進級できるかどうかの瀬戸際です。次号のベーマガが出る頃にはどうなっているのでしょう…？（岡山県津山市・SRR16歳）……【**編：**もちろん無事進級していることでしょう。**影：**そういう悩みを乗り越えて人間は大きくなるのだよ。がんばれ！】

Windows(VBScript+DirectX)用

DirectXでキレイなグラデーション

グラデーション文字(もじ)メーカー

金井　義男（38歳）

内容

入力した文字列をDirectAnimationで，グラデーション表示します。

使いかた・遊びかた

マウスの右クリックと左クリックを使って，8色の基本色からグラデーションを作成します。

メニュー・キーにより，グラデーション域の調整と位置移動，フォントサイズやタイプ・拡大・縮小・透明度・背景画像・文字列などの変更ができます。出来上がったイメージは，キャプチャ(PrtScキー)でクリップボードにbmp形式で保存したものをファイル化して使用します。

操作方法

マウスで色の変更。メニュー表示されたキーボタンで各種設定値を変更します。

プログラムの仕組み

グラデーション域の変更では文字サイズによりますが，どれぐらいの色変化量を与えるかの調整を行います。

位置移動では，グラデーション域内での位置が変更できるので，たとえば黄色から黒へのグラデーションでは，黄色を強めにするといったことがこれによって可能になります。

拡大縮小では，グラデーションの変化は起こりません。

作ったとき苦労した点

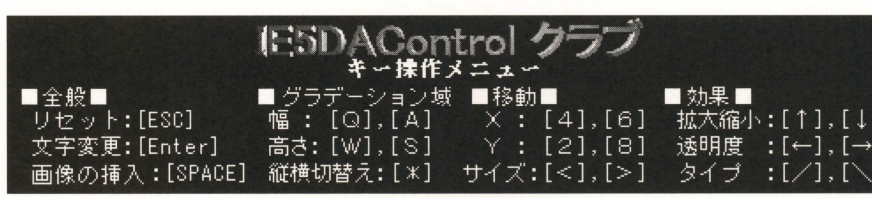
やっとビヘイビアの扱いかたが理解できました。フォントの種類を増やす方法はないでしょうか？知っている方は教えてください。

CHECKER FLAG

Dr.D：文字のクリッピングをうまく利用した作品じゃな。ソフトの説明どおりに利用するのもよいが，リスト内容を理解できる人なら“Grad_Make()”を好みの形で記述すれば，いろいろなグラデーションの2D文字列を作成できるゾ。

編：gif画像化して，ホームページのタイトル画像として使用すれば，見栄えがする形になりそうです。

つぐ美：グラデーションの色を自由に設定できるようにしても，面白いかもしれませんね。



help5.gifの画像



《写真１》カラーで見せられないのが残念な作品



《写真２》オリジナルのグラデーション文字を作ろう

★新しいホームページを制作中。名前は「NAMCO MANIAなX68000」。いまだにX68K＆NAMCOマニアを卒業できていません。（千葉県松戸市・LIZ27歳）……**【影：新食・ラーメンライスな広東料理を開発中。つぐ美：いまだにワケのわからない影さんのままです。】**

『グラデーション文字メーカー』のプログラム・リスト

```
<HTML>
<head><title>  2DGradientFont  </title></head>
<BODY BGCOLOR=WHITE TEXT=LIME LEFTMARGIN="0", TOPMARGIN="0">
<Font Size="5"Color="Red"><b>Ver1.02</b></Font>
<SCRIPT LANGUAGE="VBScript">
'---------------------------------------------------
'グローバル変数の宣言
'---------------------------------------------------
Option Explicit
  Dim m         'As DAStatics
  Dim cw        'As DANumber  クライアント ウィンドウの幅
  Dim ch        'As DANumber  クライアント ウィンドウの高さ
  Dim gt        'As Long      グラデーションタイプ(0:横縞、1:縦縞)
  Dim gtj       'As DAString  グラデーションタイプ値を漢字に変換(0→横、1→縦)
  Dim gw        'As DANumber  グラデーション横幅 (ピクセル)
  Dim gh        'As DANumber  グラデーション縦幅 (ピクセル)
  Dim gsclr     'As String    グラデーションStartColor
  Dim geclr     'As String    グラデーションEndColor
  Dim gradimg   'As DAImage   グラデーションイメージ
  Dim fnt       'As DAFontStyle  フォント
  Dim ic   'As Long 左マウスクリック時グラデーションStartColorの切り替え 0<->7
  Dim id   'As Long 右マウスクリック時グラデーションEndColorの切り替え 0<->7
  Dim ie        'As Long      右マウスクリック時フォントの切り替え0<->5
  Dim fs        'As DANumber  フォントサイズ
  Dim fn        'As DAString  フォント名
  Dim moji      'As String    クリップ文字
  Dim clipimg   'As DAImage   クリッピング文字イメージ
  Dim dispimg   'As DAImage   パラメータ文字イメージ
  Dim dispimg1  'As DAImage   パラメータ文字イメージ
  Dim messimg   'As DAImage   メッセージ文字イメージ
  Dim panimg    'As DAImage   キー操作ヘルプイメージ
  Dim backimg   'As DAImage   背景イメージ
  Dim xm        'As DANumber  X軸移動 パラメータ
  Dim ym        'As DANumber  Y軸移動 パラメータ
  Dim zm        'As DANumber  拡大縮小 パラメータ
  Dim op   'As DANumber Opacityメソッドでイメージの透明度 (0:透明～1:不透明)
  Dim fname     'As String   画像ファイル名
  Const GRAD_W = 600  'As Long  グラデーション横幅初期値
  Const GRAD_H = 200  'As Long  グラデーション縦幅初期値
  Const FONT_S = 72   'As Long  フォントサイズ初期値
  Const GRAD_S = "Yellow"  'As String  グラデーションStartColor初期値
  Const GRAD_E = "Black"   'As String  グラデーションEndColor初期値
'---------------------------------------------------
'初期化処理
'---------------------------------------------------
  Private Sub Window_onLoad()
    Dim finalimg   'As DAImage   トータルイメージ

    'メートルライブラリを使用 (右手座標系)
    Set m = DAControl.MeterLibrary
    '初期フォントタイプ
    fnt = "HGｺﾞｼｯｸE-PRO"
    '初期グラデーションタイプ (0:横、1:縦)
    gt = 1
    '初期グラデーションStart&EndColor
    gsclr = GRAD_S
    geclr = GRAD_E
    '初期クリップ文字
    moji = "GM"
    '画面表示パラメータイメージの作成
    Call Disppara()
    'グラデーションの作成
    Call Grad_Make()
    '文字のクリッピング
    Call Clip_Make()
    'メッセージ文字の作成
    Call Dispmesg()
    'キー操作ヘルプイメージの作成
    Call Dispkeyh()
    'トータル イメージの合成
    'Array関数は、配列形式のバリアント型変数を返します。
    ' (配列は、絵の重なる最上位から順に)
    Set finalimg _
    = m.OverlayArray(Array(messimg,panimg,dispimg,dispimg1,clipimg,backimg))
    DAControl.Image = finalimg
    DAControl.Start
    Call Window_onResize
  End Sub

'---------------------------------------------------
'画面サイズ変更時の再描画
'---------------------------------------------------
  Sub Window_OnResize()
    On Error Resume Next
    DAControl.Width  = Document.Body.ClientWidth
    DAControl.Height = Document.Body.ClientHeight
    'クライアントの幅と高さのビヘイビアを変更(m.Pixel = 0.000263)
    cw.SwitchToNumber Document.Body.ClientWidth * m.Pixel.Extract / 2.1
    ch.SwitchToNumber Document.Body.ClientHeight * m.Pixel.Extract / 2.1
    DAControl.focus
  End Sub

'------------------------------------------------------------
'画面表示パラメータイメージの作成
'------------------------------------------------------------
  Private Sub Disppara()
    Dim ds, ds1 'As DAString  DirectAnimation文字列
    Dim num      'As DANumber
    Dim fnt1     'As DAFontStyle  フォント
    '----------------------------
    'パラメータをビヘイビアとして定義
    '----------------------------
    Set cw = m.ModifiableNumber(0)
    Set ch = m.ModifiableNumber(0)
    Set gtj = m.ModifiableBehavior(m.DAString("横"))
    Set gw = m.ModifiableNumber(GRAD_W)
    Set gh = m.ModifiableNumber(GRAD_H)
    Set fs = m.ModifiableNumber(FONT_S)
    Set fn = m.ModifiableBehavior(m.DAString(fnt))
    Set xm = m.ModifiableNumber(0)
    Set ym = m.ModifiableNumber(0)
    Set zm = m.ModifiableNumber(1)
    Set op = m.ModifiableNumber(1)
    Set clipimg = m.ModifiableBehavior(m.EmptyImage)
    Set backimg = m.ModifiableBehavior(m.EmptyImage)  '背景の仮イメージ作成
    '----------------------------------
    '画面操作パラメータ表示イメージの作成
    '----------------------------------
    Set num = m.DANumber(100)
    'DirectAnimaitionで文字列は、DAStringオブジェクトで管理されています。
    Set ds = m.DAString("グラデーションサイズ=[W:")
    '文字列の結合 Set ds3 = m.ConcatString(ds1, ds2)
    'DANumber型の値を引数で指定された小数点桁数を持つDAString型に変換
    Set ds = m.ConcatString(ds, gw.ToString(0))
    Set ds = m.ConcatString(ds, m.DAString(" H:"))
    Set ds = m.ConcatString(ds, gh.ToString(0))
    Set ds = m.ConcatString(ds, m.DAString("]  変化方向=["))
    Set ds = m.ConcatString(ds, gtj)
    Set ds = m.ConcatString(ds, m.DAString("]  フォントサイズ=["))
    Set ds = m.ConcatString(ds, fs.ToString(0))
    Set ds = m.ConcatString(ds, m.DAString("]"))
    Set ds1 = m.DAString("移動=[X:")     'cm単位で表示する
    Set ds1 = m.ConcatString(ds1, m.Mul(xm, num).ToString(2))
    Set ds1 = m.ConcatString(ds1, m.DAString(" Y:"))
    Set ds1 = m.ConcatString(ds1, m.Mul(ym, num).ToString(2))
    Set ds1 = m.ConcatString(ds1, m.DAString("] 拡大率=["))
    Set ds1 = m.ConcatString(ds1, m.Mul(zm, num).ToString(0))
    Set ds1 = m.ConcatString(ds1, m.DAString("]% 透明度=["))
    Set ds1 = m.ConcatString(ds1, m.Mul(op, num).ToString(0))
    Set ds1 = m.ConcatString(ds1, m.DAString("]% フォント名=["))
    Set ds1 = m.ConcatString(ds1, fn)
    Set ds1 = m.ConcatString(ds1, m.DAString("]"))
    'フォントの定義
    Set fnt1 = m.DefaultFont.Family("MS ゴシック").Color(m.Green)
    'DAStringからDAImageを作成
    Set dispimg = m.StringImageAnim(ds, fnt1)
    Set dispimg1 = m.StringImageAnim(ds1, fnt1)
    '常に画面下側で表示 (ビヘイビアを使用)
    Set dispimg = dispimg.Transform(m.Translate2Anim(m.DANumber(0), m.Add(m.Mul(ch, m.DANum
ber(-1)), m.DANumber(0.005))))
    Set dispimg1 = dispimg1.Transform(m.Translate2Anim(m.DANumber(0), m.Mul(ch, m.DANumber(
-1))))
  End Sub

'------------------------------------------------------------
'グラデーションの作成
'------------------------------------------------------------
  Private Sub Grad_Make()
    Dim dxt          'As DXTMSFTLib.Pixelate
    Dim dxtres       'As IDADXTransformResult

    Set dxt = CreateObject("DXImageTransform.Microsoft.Gradient")
    dxt.GradientType = gt
    dxt.GradientWidth = gw.Extract
    dxt.GradientHeight = gh.Extract
    dxt.StartColorStr = gsclr
    dxt.EndColorStr = geclr
    Set dxtres = m.ApplyDXTransform(dxt, Null, Null)
    Set gradimg = dxtres.OutputBvr
  End Sub

'------------------------------------------------------------
'文字のクリッピング
'------------------------------------------------------------
  Private Sub Clip_Make()
    Dim matt         'As DAImage
    Dim newimg       'As DAImage

    'クリップ文字の定義
    Set matt=m.DAString(moji)
    Set matt=m.TextMatte(matt,m.DefaultFont.Family(fnt).Size(fs.Extract).Bold)
    '移動の定義
    Set matt = matt.Transform(m.Translate2Anim(xm, ym))
    '文字のクリッピング
    Set newimg = gradimg.Clip(matt)
    'イメージを拡大縮小変換
    Set newimg = newimg.Transform(m.Scale2UniformAnim(zm)) '２Ｄ拡大縮小→変換
    'イメージの透明度を変化 (0:透明～1:不透明、固定する場合はOpacityメソッド)
    Set newimg = newimg.OpacityAnim(op)     'DAImage型
    clipimg.SwitchTo newimg
  End Sub

'---------------------------------------------------
'メッセージ文字イメージの作成
'---------------------------------------------------
  Private Sub Dispmesg()
    Dim fnt2         'As DAFontStyle  フォント

'オレンジ・ＭＳ 明朝フォント・24ポイント・太字のDAFontStyleオブジェクトを作成
    Set fnt2 = m.DefaultFont.Color(m.ColorRgb255(255, 128, 0))
    Set fnt2 = fnt2.Family("ＭＳ 明朝").Size(18).Bold
    '文字列からDAImageオブジェクトを作成
    Set messimg=m.StringImage("マウス左ボタンと右ボタンで色変更します。",fnt2)
    '常に画面下側で表示 (ビヘイビアを使用)
    Set messimg = messimg.Transform(m.Translate2Anim(m.DANumber(0), m.Add(m.Mul(ch, m.DANum
ber(-1)), m.DANumber(0.01))))
  End Sub

'----------------------------
'キー操作ヘルプイメージの作成
'----------------------------
  Private Sub Dispkeyh()
    '透過色(黒)を設定してＧＩＦ画像をロード
    Set panimg = m.ImportImage("help5.gif")
    '常に画面左側で表示 (ビヘイビアを使用)
    'panimg.BoundingBox.Max.xは、help.bmpの幅の最大値?を返します。
    '一旦、中央表示から、センターがRight表示になるように移動表示し、
    Set panimg = panimg.Transform( _
    m.Translate2Anim(m.DANumber(0) , m.Sub(ch , panimg.BoundingBox.Max.y)))
  End Sub

'------------------------------------------------------------
'クリック時にグラデーションの色を変更する
'------------------------------------------------------------
  Private Sub DAControl_MouseDown(ByVal button, _
                            ByVal KeyFlags, ByVal X, ByVal Y)
    Dim newlight     'As DAGeometry
    Dim num          'As DANumber
    Dim hsl          'As DAImage 一定の時間で変化するDAColor オブジェクト

    If button = 1 Then
      'マウス左ボタンで、グラデーションStarColorの変更(ic)の
      'インクリメント (0→7 を循環)
      ic = (ic + 1) Mod 8     '適切な光源(照明)を定義
      Select Case ic
        Case 0
          gsclr = "Yellow"
        Case 1
          gsclr = "Cyan"
        Case 2
          gsclr = "Green"
        Case 3
          gsclr = "Red"
        Case 4
          gsclr = "Blue"
        Case 5
          gsclr = "Magenta"
        Case 6
          gsclr = "White"
        Case 7
          gsclr = "Black"
      End Select
    Else
      'マウス右ボタンで、グラデーションEndColorの変更(id)の
```

★とても良い本屋があった。雑誌1年間購読し続けると，1か月分がタダになる！払い込みによる定期購読より安いのでこっちを利用中。
（岡山県倉敷市・**shin15歳**）……**【編：**どこの本屋？アイデアひねってがんばってるんだね。我々もがんばろう！**】**

```
        'インクリメント (0→7 を循環)
        id = (id + 1) Mod 8    '適切な光源(照明)を定義
        Select Case id
          Case 0
            geclr = "Black"
          Case 1
            geclr = "Yellow"
          Case 2
            geclr = "Cyan"
          Case 3
            geclr = "Green"
          Case 4
            geclr = "Red"
          Case 5
            geclr = "Blue"
          Case 6
            geclr = "Magenta"
          Case 7
            geclr = "White"
        End Select
      End if
      'グラデーションの作成
      Call Grad_Make()
      '文字のクリッピング
      Call Clip_Make()
  End Sub
'-------------------------------------------------------------------
'クリッピング文字のフォントを変更する
'-------------------------------------------------------------------
  Private Sub font_chenge()
        '/\で、フォント(ie)のインクリメント (0→5 を循環)
        Select Case ie
          Case 0
            fnt = "HGゴシックE-PRO"
          Case 1
            fnt = "HG正楷書体-PRO"
          Case 2
            fnt = "HG丸ゴシックM-PRO"
          Case 3
            fnt = "MS P明朝"
          Case 4
            fnt = "MS Pゴシック"
          Case 5
            fnt = "MS 明朝"
        End Select
     fn.SwitchTo m.DAString(fnt)
  End Sub
  '--------------------------------------------------------
  'DAControl_KeyDown : キー操作に対応する処理
  '--------------------------------------------------------
  Private Sub DAControl_KeyDown(ByVal KeyCode, ByVal KeyData)
    Dim newbkimg  'As DAImage  背景イメージの貼り替え
    Dim newimg    'As DAImage

    If KeyData < 0 Then Exit Sub    'トリックコード
      'キーを押した時のDAControlのKeyDownイベントで、
      'ダイアログなどのフォーカスを持つウインドウを表示すると
      'ダイアログを閉じた後にもう一度、引数KeyDataにマイナスの値が
      '入ってイベントが発生してしまうので
      '強制的に無視するように仕込んである
    'キーコードによる分岐
    Select Case KeyCode
        '-----------------------------
        'INPUT BOX による文字列の読み込み
        '-----------------------------
        Case 13    '[Enter]
          moji = InputBox("描画作成文字列を入力","グラデーション 文字メーカー")
            If moji <> "" Then
              '文字のクリッピング
              Call Clip_Make()
            End If
        '-----------------------------
        '背景画像ファイルの読み込み
        '-----------------------------
        Case 32    '[SpaceBar]
          On Error Resume Next                'エラートラップ開始
          'コモンダイアログ コントロールの設定
          CommonDialog1.CancelError = True     '[キャンセル]ボタンでエラー
          CommonDialog1.Filter="ピクチャFiles(*.bmp;*.jpg;*.gif)|*.bmp;*.jpg;*.gif"
          CommonDialog1.FilterIndex = 1        'ファイル名のフィルタ
          CommonDialog1.ShowOpen               '[ファイルを開く]ダイアログ
            If Err = 0 Then
              'ファイル名を関数に渡してDAImageを受け取る
              fname = CommonDialog1.FileName
            ElseIf Err <> 32755 Then
              'MsgBox "Error Code No = " & Err
              'コモンダイアログ コントロールがない場合、
              'INPUT BOX によるファイル読み込みの暫定処置
              fname = InputBox("画像ファイル名を入力 [フルパスで指定]","グラデーション文字メーカー", fname)
            End If
            If fname <> "" Then 'Dir関数が使えないので、少しでもエラーを回避する
              Set newbkimg = m.ImportImage(fname)        '背景データを読み込む
              '表示イメージを差し替える
              backimg.SwitchTo newbkimg
            End if
          On Error GoTo 0                     'エラートラップ解除
        '-----------------------------
        '画面表示パラメータのリセット
        '-----------------------------
        Case 27    '[ESC]:パラメータの初期化
          Call ResetParameters()
        Case 38    '[↑]:拡大
          zm.SwitchToNumber (zm.Extract + 0.1)
        Case 40    '[↓]:縮小
          zm.SwitchToNumber (zm.Extract - 0.1)
            If zm.Extract < 0.1 Then zm.SwitchToNumber 0.1
        Case 37    '[←]:透明度(最小) = 0:透明
          op.SwitchToNumber (op.Extract - 0.01)
            If op.Extract < 0 Then op.SwitchToNumber 0
        Case 39    '[→]:透明度(最大) = 1:不透明
          op.SwitchToNumber (op.Extract + 0.01)
            If op.Extract > 1 Then op.SwitchToNumber 1
        Case 191   '[/]:フォントタイプの切り替えNEXT
          ie = (ie + 1) Mod 6
          Call font_chenge()
        Case 226   '[\]:フォントタイプの切り替えBACK
          ie = (ie - 1)
            If ie < 0 Then ie = 5
          font_chenge()
        '-----------------------------
        '移動(ポジション) 0.1m(10cm)ごと
        '-----------------------------
        Case 52, 100  '[4]:X軸  左へ移動
          xm.SwitchToNumber (xm.Extract - 0.0005)
        Case 54, 102  '[6]:X軸  右へ移動
          xm.SwitchToNumber (xm.Extract + 0.0005)
        Case 50, 98   '[2]:Y軸  下へ移動
          ym.SwitchToNumber (ym.Extract - 0.0005)
        Case 56, 104  '[8]:Y軸  上へ移動
          ym.SwitchToNumber (ym.Extract + 0.0005)
        '-----------------------------
        'グラデーションの変更
        '-----------------------------
        Case 186    '[*]:グラデーションタイプ(縦・横)変更
          gt = (gt + 1) Mod 2
            If gt = 0 Then
              gtj.SwitchTo m.DAString("横")
            Else
              gtj.SwitchTo m.DAString("縦")
            End If
        Case 81    '[Q]:X軸グラデーション作成範囲の減少
          If gw.Extract > 10 Then gw.SwitchToNumber (gw.Extract - 10)
        Case 65    '[A]:X軸グラデーション作成範囲の増加
          gw.SwitchToNumber (gw.Extract + 10)
        Case 87    '[W]:Y軸グラデーション作成範囲の減少
          If gh.Extract > 10 Then gh.SwitchToNumber (gh.Extract - 10)
        Case 83    '[S]:Y軸グラデーション作成範囲の増加
          gh.SwitchToNumber (gh.Extract + 10)
        '-----------------------------
        'フォントサイズの変更
        '-----------------------------
        Case 188    '[<]:フォントサイズの減少
          If fs.Extract > 1 Then fs.SwitchToNumber (fs.Extract - 1)
        Case 190    '[>]:フォントサイズの増加
          fs.SwitchToNumber (fs.Extract + 1)
        'Case Else
          'MsgBox "KeyCode = " & KeyCode
        End Select
        Select Case KeyCode
          Case 191, 226 , 188, 190
            '文字のクリッピング
            Call Clip_Make()
          Case 186, 81, 65, 87, 83
            'グラデーションの作成
            Grad_Make()
            '文字のクリッピング
            Call Clip_Make()
        End Select
    End Sub
'--------------------------------------------------------
'ResetParameters : 画面表示パラメータの初期化
'--------------------------------------------------------
    Function ResetParameters()
      gw.SwitchToNumber (GRAD_W)
      gh.SwitchToNumber (GRAD_H)
      fs.SwitchToNumber (FONT_S)
      xm.SwitchToNumber 0         '
      ym.SwitchToNumber 0         '
      zm.SwitchToNumber 1         '拡大縮小
      op.SwitchToNumber 1         '透明度
    End Function
</SCRIPT>

<! DirectAnimation コントロール>
  <OBJECT ID="DAControl"
    CLASSID="CLSID:B6FFC24C-7E13-11D0-9B47-00C04FC2F51D"
    STYLE="position:absolute;left:0;top:0">
  </OBJECT>
  <OBJECT ID="CommonDialog1"
    CLASSID="CLSID:F9043C85-F6F2-101A-A3C9-08002B2F49FB">
  </OBJECT>

</BODY></HTML>
```



評…そうそう、とりあえずやってみることが肝心(影)

by むつみ



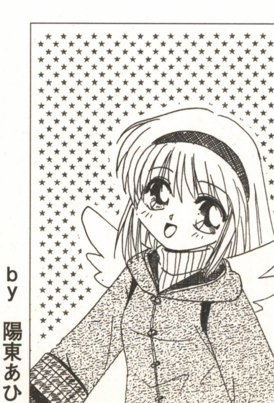
評…お昼寝ですか？私もついつい〜ぐー(影)

by 春陽



評…「漢字検定」受験した記念の投稿だそうです(編)

by 青歌

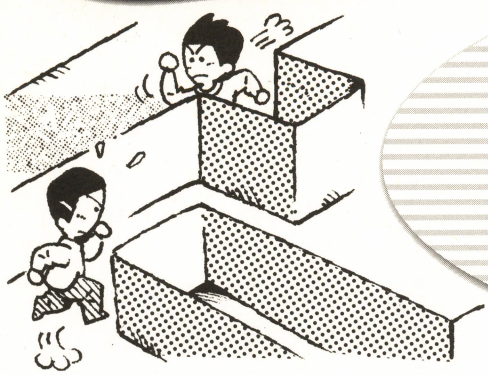


評…バックの星に文字が隠れていますって、ウッソー(影)

by 陽東あひる

★とある店で，ゲームギアがなんと！しぇいえ〜んで売ってあった。買うべきか，買わざるべきか？？（埼玉県大里郡・Mass16歳）
……【編：影さん秘蔵の「Gソニック」と「白いゲームギア」オークションに出しておきました。影：ち，ちょっと待ってよー。】

Windows(HSP V2.5)用



迷路が舞台の対戦ゲーム

@MAZE（アットメイズ）

@SYD（15歳）

内容

タイトルのとおり迷路ゲームです。迷路内に隠されたゴールを相手より先に探し出してください。2勝先取で勝ちとなります。1Pモードの場合は，勝てば勝つほどCOMの強さが上がります，というかゲームスピードが上がっていきます。負ければ元に戻ります。

《第１表》変数表

MaDa.0	迷路データ
MaDa.1	道の明るさデータ
MaDa.2	COMの通った道データ
PaDa	迷路パーツデータ
PEve	イベントの種類
PEWa	イベントのウェイト
PDam	プレイヤーのダメージ
PlFT	プレイヤーの罠探知能力の有無
PLif	プレイヤーのライフ
PlLV	プレイヤーのレベル
PlSp	プレイヤーのスピード
PMoX	プレイヤーの動く距離X
PMoY	プレイヤーの動く距離Y
PosX	プレイヤーのX座標
PosY	プレイヤーのY座標

キャラクターグラフィックは，1Pが緑色の星，2Pが黄の星にしています。また，迷路内には様々な罠（わな）が仕掛けられています。うまく潜り抜けましょう。以下は迷路に出現する罠の説明です。

※：ダメージ床です。隠れています。

♪：受けたダメージの半分回復します。

∞：相手をランダムにワープさせます。

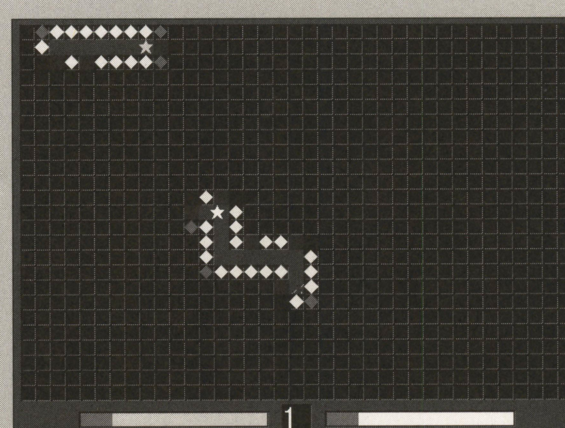
∬：スピードアップします。とればとるほど速くなります。

§：ダメージ床が見えるようになります。

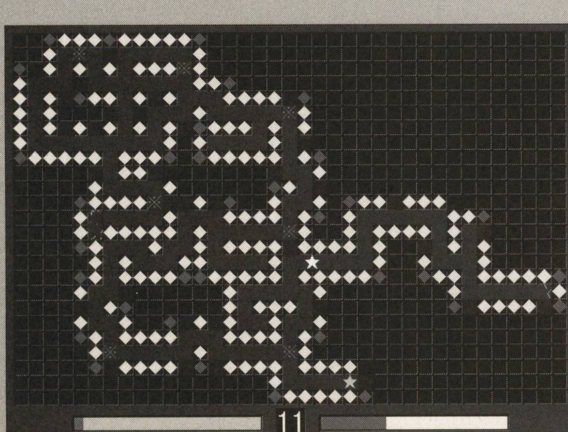
@：ゴールです。



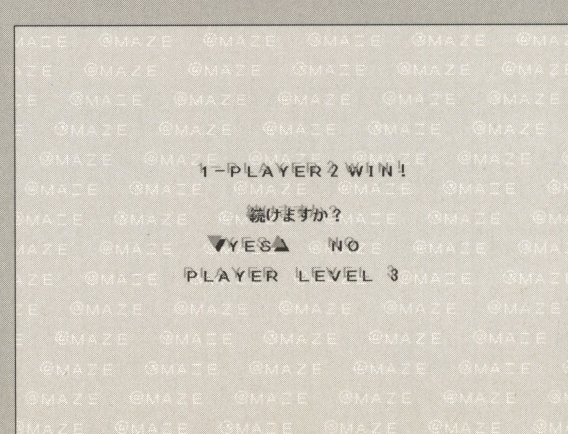
《写真1》迷路を使った対戦ゲーム



《写真2》先に迷路のゴールを探し出すのだ



《写真3》かなり大きい迷路。道の途中には罠もある



《写真4》先に2勝したほうが総合的に勝利となる

★大ピンチ！！所持金1000円きった！！！ベーマガ1冊買うのでせいいっぱい。（東京都練馬区・ルストフレム14歳）……【影：まさか，減っていく一方ってことはないでしょう。毎月お小遣いもらえるんでしょう。私なんか…。**編集長：**今月は遅刻7日分減給ね。】

その他に制限時間なんかもありますが，十分余るように設定しています。一応99カウント後にゲーム・オーバーになります。そのときは，残りライフの多いほうの勝ちとなります。

操作方法

1プレイヤーは上下右左の順に，それぞれ↑↓→←キー，2プレイヤーはWXDAキーです。また，メニューの選択は↑↓→ ←キーで行い，決定はスペース・キーです。

リスト入力について

☆が付いているサブルーチン，および○が付いている行は省略することができます。省略すると最高で，485行から245行(当社比)まで減ります。その場合は，本当にギリギリの動作になってしまいますので，手っ取り早く遊んでみたいとき以外はお勧めできません。

CHECKER FLAG

影：対戦型の迷路ゲームというアイデアは面白いね。
つぐ美：COMとの対戦は普通の迷路とは違って，先にゴールを探すという緊迫感が生まれてくるわ。
Dr.D：細かい点では，プレイ中にもゲームの終了処理がついているほうが親切な形じゃったな。また，その終了方法もタイトルでEnterキーというのは，あまり一般的ではないゾ。Escキーでの終了了が基本的じゃろう。

プログラムの仕組み

マップデータはどのように作っているのかというと，プログラムでは，まずマップを大まかに3×3に分けています。変数のPaDaに代入されているデータを見れば，パーツが6種類あることがわかると思いますが，私は，それをさらに縦横反転させて，マップデータに代入しています。だから全部で2兆6,400万とおり，スタートやゴールの位置まで考えるともう無限に近い種類があります（大袈裟?）。

あと，COMの思考ルーチンですが，これは昔のベーマガに載っていたのを参考にさせていただきました。

一応説明すると，COMは自分の歩いたマスひとつひとつに1ずつ増えていく数を代入しています。そして，より数の小さいマスを通るようにしています。わかりにくい説明だと思いますが，これで理解できる人がいれば幸いです。

最後に

ベーマガを初めて読んだのが中学1年生のとき。それからじつに4年もたってしまいました。ここにきて，ようやくプログラムを投稿できたことをうれしく思います。ちなみに，PC98版MAZEの移植バージョンなのでタイトルに“@”をつけています。いろいろ事情があり，残念ながらPC98版MAZEは投稿できませんでした。

『@MAZE』のプログラム・リスト

```
;------------------------------------------------------
;-  @m a z e    (C) @SYD 2001 all right reserved.  -
;------------------------------------------------------

        screen , , , 1                          ;☆ウインドウ初期化
        gsel , - 1
        chgdisp 3
        bgscr 2, , , 1
        gsel 2, 2
        randomize

        dim a: dim b: dim c: dim d: dim e       ;★変数初期化
        dim f: dim g: dim h: dim i: dim j: dim k
        dim PosX, 2             ;プレイヤーX座標
        dim PosY, 2             ;プレイヤーY座標
        dim PMoX, 2             ;動く距離X
        dim PMoY, 2             ;動く距離Y
        dim PLif, 2             ;プレイヤーライフ
        dim PDam, 2             ;プレイヤーダメージ
        dim P1Sp, 2             ;プレイヤースピード
        dim P1FT, 2             ;プレイヤーファインドトラップ
        dim PEve, 2             ;イベントナンバー
        dim PEWa, 2             ;イベントウェイト
        dim TPos                ;タイトル座標
        dim PaNu                ;パレットナンバー
        dim GMod                ;ゲームモード
        dim EMod                ;エンドモード
        dim GTim                ;ゲームタイム
        dim GamW                ;ゲームウェイト
        dim IKey, 4             ;キーコード
        dim P1Wi                ;勝ちプレーヤー
        dim PWNu, 3             ;勝ち数
        dim P1LV                ;プレイヤーレベル
        sdim PaDa, 69, 7        ;パーツデータ
        PaDa.0 = "10111110001,11111112101,11110111111,11110111001,01111111111,01110101110"
        PaDa.1 = "10100012111,20001010101,00010101000,00010101111,11010000001,11011100010"
        PaDa.2 = "10110110500,11111111111,11111201111,11112101000,00010111001,10000010012"
        PaDa.3 = "10000100101,10101000101,10000100001,10000001111,11110002111,10111160101"
        PaDa.4 = "11111201101,11121111101,00012111101,12110110001,10011101000,11200000101"
        PaDa.5 = "10000100001,10101010201,11110000101,00010010001,10000001001,00111100101"
        PaDa.6 = "11110111111,70111111111,10000111101,41110111111,21111111111,11100111111"

        repeat 32                               ;☆色初期化
                palette cnt, cnt * 8, cnt * 8, cnt * 8          ;黒>白
                palette cnt + 32, , cnt * 8, cnt * 8            ;黒>水
                palette cnt + 64, cnt * 8, 255, 255             ;水>白
                palette cnt + 96, 255, cnt * 4 + 128, cnt * 8   ;赤>白
        loop
        palette 128, 42         ;赤
        palette 129, 84         ;赤
        palette 130, 126        ;赤
        palette 131, 255        ;赤
        palette 132, 255, , 255 ;紫
        palette 133, , 255      ;緑
        palette 134, 255, 255   ;黄
        palette 135, , , 255    ;青
        palfade

        gosub Logo              ;○ロゴ
*Flag1
        gosub MazeTitle         ;○タイトル
*Flag2
        gosub MapData           ;●マップデータ初期化
        gosub PlayerData        ;●プレイヤーデータ初期化
        gosub BattleScreen      ;○バトルスクリーン
        gosub MainLoop          ;●メインループ
        gosub BattleWin         ;○バトル勝利
        if EMod = 0: goto Flag2
        gosub EndTitle          ;○エンド>タイトル
        goto Flag1

*Logo                                           ;☆ロゴ
        wait 200

        repeat 32
                palcolor - cnt + 31
                pos 232, 232
                mes "@SYD  PRODUCE"
                await 33
        loop

        wait 200

        repeat 32
                PaNu = cnt
                redraw 2
                gosub TitleBackGround   ;○タイトル背景
                palcolor PaNu + 32
                pos 232, 232
                mes "@SYD  PRODUCE"
                gosub TitleMenu         ;○タイトルメニュー
                redraw 1
                await 33
        loop
        return

*MazeTitle                                      ;☆タイトル
        repeat
                redraw 2
                PaNu = 31
                gosub TitleBackGround   ;○タイトル背景
                gosub TitleMenu         ;○タイトルメニュー
                redraw 1
                await 33
                stick IKey
                if IKey&10: GMod = IKey - 2 / 6
                if IKey = 16: break
                if IKey = 32: end
        loop
        return

*TitleBackGround                                ;☆タイトル背景
        palcolor - PaNu + 95
        boxf 0, 0, 639, 479
        palcolor 31
        sysfont
        repeat 16
                a = cnt
                repeat 9
```


★この頃どうもおかしい。作ろうとするプログラムのネタを先取りされまくっている（——；）。さては…！（新潟市・whitefox18歳）
……【影：さては，私がwhitefoxくんのパソコンに忍び込んで，ネタを持ち去っているとでも言いたいんだろ。オチまで先取りだ。】

```
			pos - TPos * 3 - (a * 16) + (cnt * 112) , - TPos + (a * 32)
			mes "@MAZE"
		loop
	loop

	if TPos = 31: TPos = 0: else: TPos++
	return

*TitleMenu					;☆タイトルメニュー
	repeat 2
		font "MS 明朝", 64
		palcolor cnt * 96 - PaNu + 31
		pos - cnt * 4 + 165, - cnt * 4 + 173
		mes "@MAZE"
		font "MS ゴシック", 16, 1
		pos - cnt * 4 + 240 , - cnt * 4 + 264
		mes "1-PLAYER vs COMPUTER¥n ¥n1-PLAYER vs 2-PLAYER"
		pos - cnt * 4 + 224, GMod * 8 - cnt * 4 + 264
		mes "▼                    ▲"
	loop
	P1LV = 0
	return

*MapData					;★マップデータ初期化
	dim MaDa, 3, 37, 25

	repeat 25
		a = cnt
		repeat 37
			MaDa.2.cnt.a = 999999999
		loop
	loop

	repeat 2
		a = cnt
		repeat 3
			b = cnt * 12 + 6: c = a * 8 + 8
			MaDa.0.b.c = 1
			rnd MaDa.2.b.c, 10: MaDa.2.b.c - 10
			b = a * 12 + 12: c = cnt * 8 + 4
			MaDa.0.b.c = 1
			rnd MaDa.2.b.c, 10: MaDa.2.b.c - 10
		loop
	loop

	repeat 3
		a = cnt
	repeat 3
		b = cnt
		rnd c, 6: rnd d, 2: rnd e, 2
		repeat 7
			f = cnt
			if d: g = cnt: else: g = 6 - cnt
			repeat 11
				if e: h=c*12+cnt: else: h=c*12-cnt+10
				strmid i, PaDa.g, h, 1
				int i
				j = a * 12 + cnt + 1: k = b * 8 + f + 1
				MaDa.0.j.k = i
				rnd MaDa.2.j.k, 10: MaDa.2.j.k - 10
				if MaDa.0.j.k = 2: MaDa.2.j.k = 0
			loop
		loop
	loop
	loop
	return

*PlayerData					;★プレイヤーデータ初期化
	repeat 3
		repeat
			rnd b, 37: rnd c, 25
			if MaDa.0.b.c = 1: break
		loop
		if cnt = 2 {
			MaDa.0.b.c = 8: MaDa.2.b.c = - 20
		}
		else {
		    PosX.cnt = b: PosY.cnt = c: PMoX.cnt = 0: PMoY.cnt = 0
		    PLif.cnt = 200: PDam.cnt = 0: P1Sp.cnt = 1: P1FT.cnt = 0
		}
	loop
	P1Wi = 0
	return

*BattleScreen					;☆バトルスクリーン
	redraw 2

	palcolor 0
	boxf 24, 24, 631, 471
	palcolor 135
	boxf 16, 16, 623, 463
	palcolor 0
	boxf 24, 24, 615, 423

	repeat 25
		a = cnt
		repeat 37
			b = cnt * 16 + 25: c = a * 16 + 25
			d = cnt * 16 + 39: e = a * 16 + 39
			gosub BattleBackGround	;○バトル背景
		loop
	loop

	repeat 2
		b = cnt * 264 + 86: c = 435
		d = cnt * 264 + 289: e = 452
		gosub BattleBackGround		;○バトル背景
		palcolor 131
		boxf b + 2, c + 2, d - 2, e - 2
	loop

	b = 304: c = 428: d = 335: e = 459
	gosub BattleBackGround			;○バトル背景

	repeat 240
		redraw 1, , cnt * 2, 639, 1
		redraw 1, , - cnt * 2 + 479, 639, 1
		await 8
	loop
	return

*BattleBackGround				;☆バトル背景
	palcolor 0
	boxf b, c, d, e
	palcolor 16
	boxf b + 1, c + 1, d, e
	palcolor 0
	boxf b + 1, c + 1, d - 1, e - 1
	return

*MainLoop					;★メインループ
	repeat
		GTim = cnt
		gosub Player1		;●1Pキー入力
		gosub Player2		;●2Pキー入力
		gosub Charactor		;●道筋表示
		gosub PlayerLife	;●ライフ表示
		if P1Wi: break
		await - GamW + 16
	loop
	return

*Player1					;★1Pキー入力
	tr = 0
	if PMoX.0 = PMoY.0 {
		stick IKey, 15
		PMoX.0 = (IKey = 4) - (IKey = 1)
		PMoY.0 = (IKey = 8) - (IKey = 2)
		gosub Move		;●キャラ移動
	}
	return

*Player2					;★2Pキー入力
	tr = 1
	if PMoX.1 = PMoY.1 {
		if GMod {
			await 0
			getkey IKey.0, 87: getkey IKey.1, 68
			getkey IKey.2, 88: getkey IKey.3, 65
			PMoX.1 = IKey.1 - IKey.3: PMoY.1 = IKey.2 - IKey.0
			if (PMoX.1 ! 0) & (PMoY.1 ! 0): return
			gosub Move	;●キャラ移動
		}
		else {				;★COM移動設定
			a = PosX.1 - 1: b = PosY.1: gosub Check: c=MaDa.2.a.b
			a = PosX.1 + 1: gosub Check: d = MaDa.2.a.b
			a = PosX.1: b = PosY.1 - 1: gosub Check: e=MaDa.2.a.b
			b = PosY.1 + 1: gosub Check: f = MaDa.2.a.b
			if (c < d) & (c < e) & (c < f) : PMoX.1 = - 1: else {
				if (d < e) & (d < f) : PMoX.1 = 1: else {
					if (e < f) : PMoY.1=-1: else: PMoY.1=1
				}
			}
			gosub Move	;●キャラ移動
			MaDa.2.a.b = GTim
		}
	}
	return

*Charactor					;★道筋表示
	repeat 2
		redraw 2
		tr = cnt
	repeat 3
		a = cnt
	repeat 3
		b=PosX.tr+cnt-1: c=PosY.tr+a-1: d=(cnt=1)+(a=1)+1
		if MaDa.1.b.c < d {
			pos b * 16 + 24, c * 16 + 24
			if MaDa.0.b.c ! 0 {
				palcolor d + 127
				boxf b*16+24, c*16+24, b*16+39, c*16+39
			}
			if MaDa.0.b.c = 0: palcolor d * 15: mes "◆"
			MaDa.1.b.c = d
		}
		if (P1FT.tr = 1) & (MaDa.0.b.c = 2): MaDa.0.b.c++
		palcolor 132
		pos b * 16 + 24, c * 16 + 24
		if MaDa.0.b.c = 3: mes "※"
		palcolor 65
		if MaDa.0.b.c = 4: mes "♪"
		if MaDa.0.b.c = 5: mes "∞"
		if MaDa.0.b.c = 6: mes "∬"
		if MaDa.0.b.c = 7: mes "§"
		if MaDa.0.b.c = 8: mes "@"
	loop
	loop
	loop

	repeat 2				;★キャラ表示
		tr = cnt
		redraw 2
		palcolor 130
		pos PosX.tr * 16 - PMoX.tr + 24, PosY.tr * 16 - PMoY.tr + 24
		mes "★"
		repeat P1Sp.tr
			if PMoX.tr > 0: PMoX.tr--: gosub Event: else {
				if PMoX.tr < 0: PMoX.tr++: gosub Event
			}
			if PMoY.tr > 0: PMoY.tr--: gosub Event: else {
				if PMoY.tr < 0: PMoY.tr++: gosub Event
			}
		loop
		redraw 2
		if PEWa.tr ¥ 2: palcolor 255: else: palcolor tr + 133
		if PEWa.tr = 59: palcolor 130
		pos PosX.tr * 16 - PMoX.tr + 24, PosY.tr * 16 - PMoY.tr + 24
		mes "★"
		redraw 1, PosX.tr * 16 + 8, PosY.tr * 16 + 8, 48, 48
	loop
	return

*Event						;★イベントサーチ
	if PMoX.tr ! PMoY.tr: return
	a = PosX.tr: b = PosY.tr
	if MaDa.0.a.b = 2: MaDa.0.a.b = 3
	if MaDa.0.a.b / 4 = 1 {
		PEve.tr = MaDa.0.a.b - 3
		MaDa.0.a.b = 1
	}
	if MaDa.0.a.b = 3: PLif.tr - 10		;床ダメージ
	if MaDa.0.a.b = 8: P1Wi = tr + 1
```


★なんと，学校の生協にベーマガが戻りました。今まで即刻売り切れていただけだったのかも……。って，え？つぐ美さんが？！
(東京都狛江市・kijimon19歳) ……【つぐ美：じつはこっそり私が生協にお願いしておいたのよ。影：単独行動は危険ですぜ，姉御。】

```
        return

*PlayerLife                                     ;★ライフ表示
        redraw 2

        repeat 2
                tr = cnt
                if PDam.tr > PLif.tr {
                        PDam.tr--
                        palcolor 131
                        line tr*264-PDam.tr+287, 436, tr*264-PDam.tr+287, 450
                }
                if PDam.tr < PLif.tr {
                        PDam.tr++
                        palcolor tr + 133
                        line tr*264-PDam.tr+288, 436, tr*264-PDam.tr+288, 450
                }
                if PDam.tr = 0: P1Wi = 2 - tr
        loop

        palcolor                                ;★残りタイム表示
        boxf 305, 429, 334, 458
        a = GTim/120
        if a = 100 : a = 99 : P1Wi = (PLif.1 > PLif.0) + 1
        font "MS ゴシック",32
        palcolor 31
        pos 304, 428
        mes a
        sysfont

        redraw 1, 87, 429, 552, 457
        return

*Move                                           ;★キャラ移動
        if PEve.tr = 1: goto Cure       ;☆回復
        if PEve.tr = 2: goto Warp       ;☆ワープ
        if PEve.tr = 3: goto SpeedUp    ;☆スピードアップ
        if PEve.tr = 4: goto FindTrap   ;☆ファインドトラップ

        a = PosX.tr + PMoX.tr
        b = PosY.tr + PMoY.tr
        if MaDa.0.a.b = 0 {
                PMoX.tr = 0: PMoY.tr = 0
                return
        }
        PosX.tr = a
        PosY.tr = b
        PMoX.tr = PMoX.tr * 16
        PMoY.tr = PMoY.tr * 16
        return

*Check                                          ;★COMのLIFEチェック
        if PLif.tr >10: return
        if MaDa.0.a.b / 2 = 1: MaDa.2.a.b = 999999999
        return

*Cure                                           ;☆回復
        PEWa.tr++
        if PEWa.tr = 1: PLif.tr += 200 - PLif.tr / 2
        if PEWa.tr = 120: PEve.tr = 0: PEWa.tr = 0
        return

*Warp                                           ;☆ワープ
        if PEWa.tr = 0 {
                PEve.tr = 0
                c = 1 - tr: PEve.c = 2: PEWa.c = 1
                return
        }
        PEWa.tr++
        if PEWa.tr = 60 {
                repeat
                        rnd c, 37: rnd d, 25
                        if MaDa.0.c.d = 1: break
                loop
                PosX.tr = c: PosY.tr = d
        }
        if PEWa.tr = 120: PEve.tr = 0: PEWa.tr = 0
        return

*SpeedUp                                        ;☆スピードアップ
        PEWa.tr++
        if PEWa.tr = 120: PEve.tr = 0: PEWa.tr = 0: P1Sp.tr++
        return

*FindTrap                                       ;☆ファインドトラップ
        PEWa.tr++: P1FT.tr = 1
        if PEWa.tr = 120: PEve.tr = 0: PEWa.tr = 0
```

```
        return

*BattleWin                                      ;☆バトル勝利
        wait 100
        repeat 32
                a = cnt
                redraw 2
                palcolor a
                repeat 15
                        b = cnt
                        repeat 20
                                boxf cnt*32, b*32, cnt*32+a, b*32+a
                        loop
                loop
                redraw 1
                await 33
        loop

        EMod = 0
        PWNu.P1Wi++     ;勝ち数
        if P1Wi = 1: P1LV ++: else: P1LV --
        if PWNu.1 = 2: P1LV + GamW / 2 + 1

        repeat 32
                PaNu = cnt
                redraw 2
                gosub TitleBackGround   ;☆タイトル背景
                gosub EndMenu           ;☆エンドメニュー
                redraw 1
                await 33
        loop

        if PWNu.P1Wi = 2 {
                repeat
                        stick IKey
                        if IKey&5: EMod = (IKey = 4)
                        if IKey&16: break
                        redraw 2
                        PaNu = 31
                        gosub TitleBackGround   ;☆タイトル背景
                        gosub EndMenu           ;☆エンドメニュー
                        redraw 1
                        await 33
                loop

                if (GMod = 1) | (EMod = 1) | (PWNu.1 ! 2): GamW = 0: P1LV = 0: else {
                        if GamW < 15 : GamW + 2
                }
                PWNu.1 = 0, 0
        }
        return

*EndMenu                                        ;☆エンドメニュー
        repeat 2
                font "MS ゴシック", 16, 1
                palcolor cnt * 96 - PaNu + 31
                pos - cnt * 4 + 221, - cnt * 4 + 180
                if P1Wi = 1: mes "1-PLAYER"
                if (P1Wi = 2) & (GMod=0): mes "COMPUTER"
                if (P1Wi = 2) & (GMod=1): mes "2-PLAYER"
                pos - cnt * 4 + 365, - cnt * 4 + 180
                mes "" + PWNu.P1Wi + " WIN!"
                if PWNu.P1Wi = 2 {
                        pos - cnt * 4 + 252, - cnt * 4 + 228
                        mes "  続けますか?¥n ¥nYES      NO"
                        pos EMod * 24 - cnt * 4 + 236 , - cnt * 4 + 260
                        mes "▼          ▲"
                        pos - cnt * 4 + 208 , - cnt * 4 + 292
                        if GMod = 0: mes "PLAYER LEVEL " + P1LV
                }
        loop
        return

*EndTitle                                       ;☆エンド>タイトル
        repeat 32
                redraw 2
                PaNu = 31
                gosub TitleBackGround   ;☆タイトル背景
                PaNu = - cnt + 31
                gosub EndMenu           ;☆エンドメニュー
                PaNu = cnt
                gosub TitleMenu         ;☆タイトルメニュー
                redraw 1
                await 33
        loop
        return
```

評…高校合格おめでとう！記念に掲載です（編）



by　IWACHAN

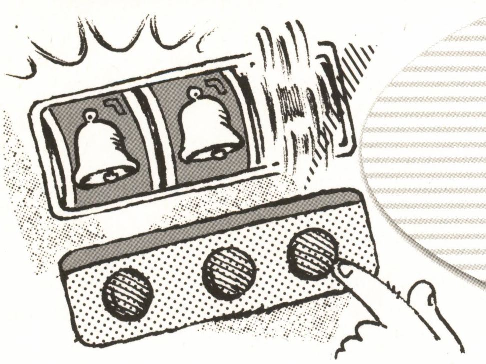
評…久しぶりですね，これからも投稿よろしく（つぐ美）



by　比呂

★ぷー。①屁　②とことんぷーたろー　③どせいさん　さて，何でしょう？（長野県松本市・TAC25歳）……【影：へぇー　①編くんが私の発言に感心している　②つぐ美ちゃんが私の発言を聞き流している　③私の発言に自分で合いの手を入れている　さて何？】

Java(JDK1.1)用



javaアプレットプログラムの登場

JavaSlot

ジャバ スロット

チュにひろ（19歳）

内容

javaアプレットによるスロット・ゲームです。真ん中の横一列に同じ模様を揃えるようにしてください。三つのドラムはそれぞれ違うスピードで回転するようにしてみました。動作確認はJDK1.1のappletviewerならびにInternetExplorer5で確認しています。

遊びかた

リストをコンパイルしてできた各classファイルとslot.htmを同じディレクトリに置いてから，slot.htmを実行させます。

ゲームはSTARTボタンを押してドラムを回転させ，回転する三つのドラムを対応するstopを押して止めてください。

入力方法

まず，Javaのプログラムを始めるには，JDKというツールを入手する必要があります。このJDKはサン・マイクロシステムズのホームページからも入手できますが，個人的にはJavaを扱っている雑誌やJavaの入門書についてくるCD-ROMからの入手がオススメ。

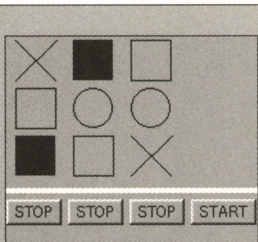
掲載されているリストをテキストエディタ(メモ帳でもOK。ただ，入力はチョットやりにくい)で入力し，指定されているファイル名で保存します。このファイル名を間違えると，正しくコンパイルされないことがあります。

入力し終えたら，JDKにある“javac.exe”を使ってコンパイルを行います。コンパイル方法は，MS-DOSプロンプトから“javac javaslot.java”などと入力します。正しくコンパイルされると，ファイル名.classというファイルが生成されます。この動作をjavaslot.java，buttons.javaの二つのファイルに対して行ってください。

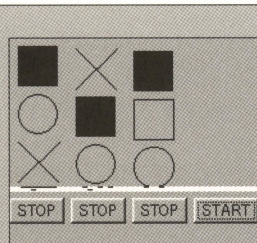
slot.htmはテキストエディタでリストどおりに入力して，拡張子を.htmにして

《第１表》変数表

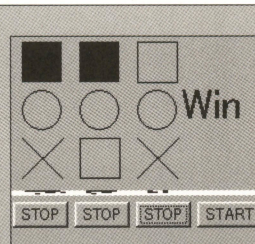
int[][] slot	各スロットドラムの内容 (0:○，1:×，2:□，3:■)
int[][] y	各スロットドラムのy座標
int[] sp	各スロットドラムのスピード
int[] stop	各スロットドラムのストップ判定フラグ



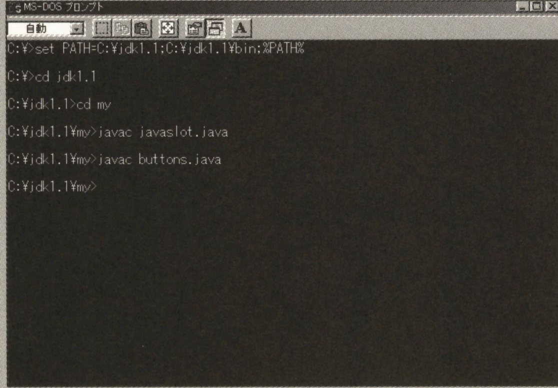
《写真１》ゲームの基本の一つ，スロットマシーン



《写真２》それぞれのドラムの回転速度が違う点に注目



《写真３》真ん中の横一列に同じ模様を並べるようにする



《写真４》リストのコンパイルはDOSプロンプトで行う

CHECKER FLAG

編：javaアプレットプログラムの採用です。パソコン上で動作する形ですが，このプログラムの掲載をきっかけにiモード対応のjavaプログラムが投稿されるかもしれませんね。

Dr.D：iモード対応のjavaでプログラムを始めるために，まずはこういった短めの作品から，javaプログラミングの内容を把握してほしいものじゃな。

つぐ美：初めて採用するプログラム言語ということで，写真４にコンパイル画面を載せてみました。参考にしてくださいね。javacへは“set PATH=C:¥jdk1.1;C:¥jdk1.1¥bin;%PATH%”という感じでパスを通しておくといいわよ。

保存すればOKです。このファイルはコンパイルは行いません。

プログラムの仕組み

javaslot.javaで主にゲーム本体の処理を行っています。画面を定期的に更新していくため，このプログラムではスレッドを使って画面の更新を行っています。buttons.javaはSTART，STOPボタンを配置しています。そしてこの二つをアプレット間の通信機能を使って動作が連動するようにしています。

コメント

javaプログラムではあるけど，内容は単純。それでもスレッドやアプレット通信を使ったりしているので，リスト面ではチョット工夫があると言えるかも？ 個人的にはjavaプログラムは，まだ慣れていないので組みかたに問題があるかも? リストの容量は比較的短いと思うんだけど，どんなもんでしょうか?

《リスト1》javaslot.javaのプログラム・リスト

```
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;

public class javaslot extends Applet implements Runnable{
	private int[][] slot = {{0,1,2,3},{1,3,0,2},{3,2,0,1}};
	private int[][] y = new int[3][4];
	private int[] sp = new int[3];
	private int[] stop = new int[3];
	private Thread mThread = null;

	//フィールド(変数)初期化
	public void init(){
		for(int i=0;i<3;i++){
			stop[i]=0;
			for(int j=0;j<4;j++)
				y[i][j]=-32+(32*j)+(j*5);
		}
	}

	//スレッド(ゲーム)開始
	public void start(){
		if(mThread == null){
			mThread = new Thread(this, "slot");
			mThread.start();
		}
	}

	//スレッド(ゲーム)ストップ
	public void stop(){
		mThread = null;
	}

	//スレッド(ゲーム)実行
	public void run(){
		while(Thread.currentThread() == mThread){
			repaint();
			try{
				mThread.sleep(50);
			}catch(InterruptedException e){}
		}
	}

	//画面描画
	public void paint(Graphics g){
		int f1=0,f2=0;
		for(int i=0;i<3;i++){
			for(int j=0;j<4;j++){
				slotdraw(i,j,g);
			}
		}

		//STOPボタンが3つ押されたかチェック
		for(int i=0;i<3;i++){
			if(stop[i]==1)
				f1++;
			if(sp[i]==0)
				f2++;
		}
		if(f1==3){
			if(f2==3)
				hantei(g);
		}
	}

	//スロットを描画するメソッド(関数)
	public void slotdraw(int i, int j, Graphics g){
		int x = i*45+10;
		int my;

		switch(slot[i][j]){
			case 0:
				g.drawOval(x,y[i][j],31,31);
				break;
			case 1:
				g.drawLine(x,y[i][j],x+31,y[i][j]+31);
				g.drawLine(x+32,y[i][j],x,y[i][j]+31);
				break;
			case 2:
				g.drawRect(x,y[i][j],31,31);
				break;
			case 3:
				g.fillRect(x,y[i][j],31,31);
				break;
		}
		y[i][j]+=sp[i];
		if(y[i][0]>=0)
			swap(i);
	}
```


★連載コーナーがばっさり切られたことより，プレゼントがなくなったことにショックな私はダメですか？（岩手県釜石市・鉄人坂井23歳）……【編：プレゼントページはなくなりましたが，今月のOFコーナー最後を見てください。こんな感じでプレゼント再開します。】

```
	//STARTボタンが押された時の処理
	//initメソッドの呼び出し&スロットスピード設定
	public void slotstart(){
		int j=0,n=0;

		init();
		for(int i=0;i<3;i++)
			sp[i]=(i*2)+6;
	}

	//スロット判定配列の入れ替え
	public void swap(int i){
		int temp;

		temp=slot[i][3];
		slot[i][3]=slot[i][2];
		slot[i][2]=slot[i][1];
		slot[i][1]=slot[i][0];
		slot[i][0]=temp;
		for(int j=0;j<4;j++){
			y[i][j]=-32+(32*j)+(j*5);
		}
		if(stop[i]==1)
			sp[i]=0;
	}

	//STOPボタンが押された時の処理
	//STOP配列に1を代入&スロットスピードを落とす
	public void slotstop(int i){
		if(sp[i]!=0){
			sp[i]=2;
			stop[i]=1;
		}
	}

	//揃っているかを判定するメソッド(関数)
	public void hantei(Graphics g){
		int f=0,num;

		num=slot[0][2];
		for(int i=1;i<3;i++){
			if(num==slot[i][2])
				f++;
		}
		if(f==2){
			g.setFont(new Font("TimeRoman",Font.PLAIN,32));
			g.drawString("Win",135,64);
		}
	}
}
```

《リスト2》buttons.javaのプログラム・リスト

```
import java.awt.*;
import java.awt.event.*;
import java.applet.Applet;

public class buttons extends Applet{

		//アプレット通信設定&ボタンをセット
	public void init(){
		String targetname = getParameter("targetname");
		Applet target = getAppletContext().getApplet(targetname);
			for(int i=0; i<3; i++){
				Button b = new Button("STOP");
				b.addActionListener(new MAListener(target,i));
				add(b);
			}
			Button start = new Button("START");
			start.addActionListener(new MAListener(target,4));
			add(start);
	}

		//ボタンが押された時のアクション処理
		private class MAListener implements ActionListener{
			Applet mApplet;
			int num;

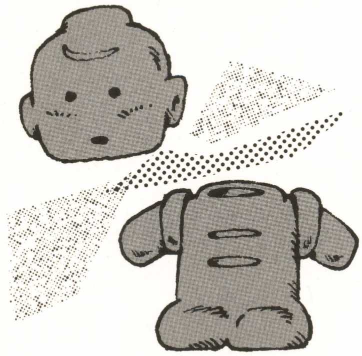
			public MAListener(Applet app, int n){
				mApplet = app;
				num = n;
			}
			public void actionPerformed(ActionEvent e){
				if(num<3)
					((javaslot)mApplet).slotstop(num);
				if(num==4)
					((javaslot)mApplet).slotstart();
			}
		}
}
```

《リスト3》slot.htmのプログラム・リスト

```
<html>
<head>
<title>Java Program</title>
</head>
<body>
<applet code="javaslot.class" width=200 height=120 name="javaslot">
</applet>
<br>
<applet code="buttons.class" width=200 height=55 name="button">
<param name="targetname" value="javaslot">
</applet>
</body>
</html>
```

★最近，パソコンが突然再起動するようになってしまった。（北海道空知郡・**WILDWOLF19歳**）……**【影：**じつは私のパソコンもそうなんですよ。だから…。**編集長：**仕事の遅れをパソコンのせいにするのだけはやめような，影くん。**影：**い，いつの間に！！**】**

PC-9801シリーズ互換機(N88-日本語BASIC(86)DOS-BASIC)用



自分の頭を武器に敵を倒せ

頭(あたま)をつかえ！

Tomrizer（21歳）

内容

頭をつかうゲームです，頭をつかうといってもパズルゲームではありません。主人公は頭を飛ばすことができます。頭と体がはずれる不気味な主人公です。飛ばした頭を敵にぶっつけてやっつけるゲームです。

遊びかた・操作方法

4, 6 キーで左右移動，8 キーでジャンプ，スペース・キーで頭をとばします。頭はジャンプした勢いでとばすので，ジャンプしているときにスペース・キーを押さないと飛びません。頭を敵に当てて倒してください。敵は全部で20体出現します。

主人公の頭は完全無敵だが体は弱いので，体が敵に当たるとゲーム・オーバーです。頭が地面に落ちたら拾ってください。得点は頭を空中で連続で敵に当てると上がっていきます。頭が空中に飛んでいるときは4, 6キーで少しだけ左右に制御できます。

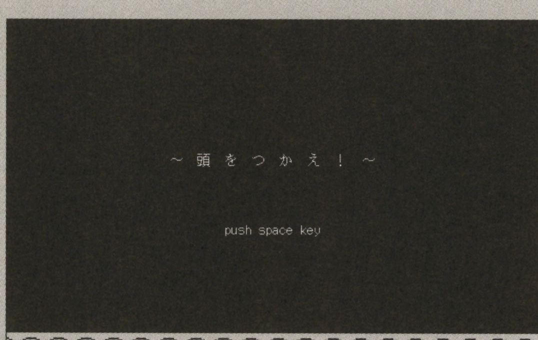
《第１表》変数表

変数	内容
SCORE	スコア
HISC	ハイスコア
X	主人公のX座標
Y	主人公のY座標
VX	主人公のX座標移動量
VY	主人公のX座標移動量
AT	頭が体からはなれてるかどうか
AX	頭のX座標
AY	頭のY座標
AVX	頭のX座標移動量
AVY	頭のY座標移動量
RS	空中であたった回数
EN	敵の数
E	敵の状態
EX	敵X座標
EY	敵Y座標
EVX	敵のX座標移動量
EVY	敵のY座標移動量
EG	敵をたおした数

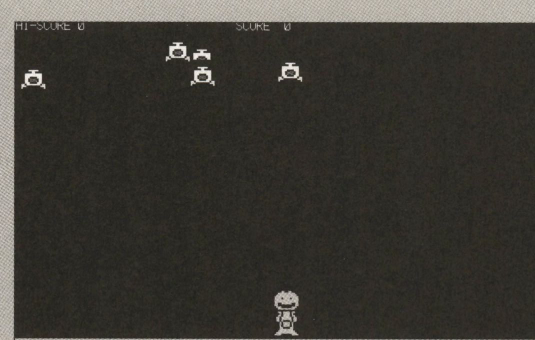
CHECKER FLAG

編：敵を倒していく基本的なアクションですが，自キャラの頭を飛ばすというのが面白いですね。初めて見たときのインパクトはバツグンのゲームです。

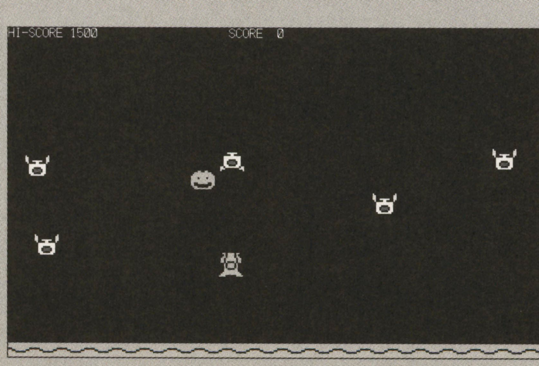
Dr.D：内容は基本的でも，アクションにちょっとしたユーモアを盛り込むだけでもゲームは面白くなってくる，という良い見本じゃな。こういうアイデアは評価できるゾ。



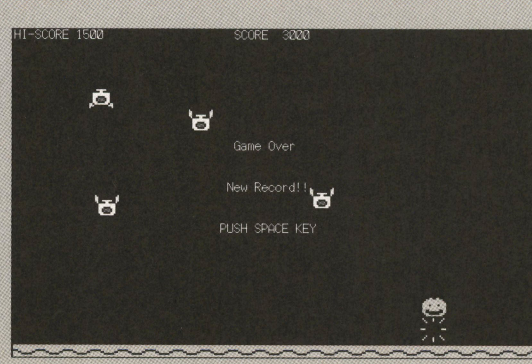
《写真１》プレイヤーが頭を使うわけではない



《写真２》普通のアクション・ゲームみたいだが…



《写真３》主人公は頭を飛ばして敵を倒す



《写真４》逆に体がやられるとゲーム・オーバー

★おや，Internet読者ハガキでは，あくせすROOM宛てのものは送れないんですか？（新潟市・佐藤良秋20歳）……【編：すみません，項目に「あくせすROOM」がぬけてましたね。OF，質問，感想の欄に，あくせすROOM宛てだとわかるように書いてください。】

『頭をつかえ！』プログラム・ソフト

```
10 SCREEN 3,0,0,1:COLOR ,,,,2:DIM G%(257,6)
20 CONSOLE 0,25,0,1:GOSUB *MAKECHR
30 RANDOMIZE VAL(RIGHT$(TIME$,2)):OUT &H77,&H76
40 '-----------------------------------------------------------------
50 GOSUB *GAMEN:GOSUB *TITLE
60 WHILE 1:GOSUB *INIT:CLS
70 FOR I=0 TO 4:E(I)=1:EX(I)=0:EY(I)=0:NEXT
80 GF=1:WHILE GF
90 GOSUB *IN
100 GOSUB *KARADA:GOSUB *ATAMA :GOSUB *TEKI
110 LOCATE 32,0:PRINT"SCORE ";SCORE:LOCATE 0,0:PRINT"HI-SCORE";HISC
120 OUT &HA6,SC:SC=SC XOR 1:OUT &HA4,SC
130 LINE(0,0)-(639,366),0,BF
140 WEND
150 GOSUB *GAMEOVER:WEND
160 '--------------------画面-------------------------------------
170 *GAMEN
180 FOR I=0 TO 1:OUT &HA6,I:CLS 3
190 FOR J=0 TO 19:PUT(J*32,367),G%(0,6),PSET:NEXT
200 NEXT:RETURN
210 '--------------------入力（移動、ジャンプ、頭飛ばし）--------
220 *IN
230 IF GM=1 THEN GOTO 290
240 IN0=INP(&HE0):IN1=INP(&HE1):VX=0
250 IF (IN0 AND &H10)=0 THEN VX=-2:AS=AS+1:IF AS>10 THEN AS=0:AN=AN XOR 1
260 IF (IN0 AND &H40)=0 THEN VX=2 :AS=AS+1:IF AS>10 THEN AS=0:AN=AN XOR 1
270 IF (IN1 AND &H1)=0 AND Y=335 THEN VY=-4
280 IF INP(&HE9)=191 AND AT<>1 THEN AT=1:AX=X:AY=Y-30:AVX=VX:AVY=VY
290 RETURN
300 '--------------------初期化-----------------------------------
310 *INIT
320 X=320:Y=335:VX=0:VY=0:AX=320:AY=336:AVX=0:AVY=0:GM=0:AT=0:SW=0:SCORE=0
330 EN=0:EG=0:FOR I=0 TO 4:E(I)=0:EX(I)=0:EY(I)=0:DF(I)=0:NEXT
340 RETURN
350 '----------------------体-------------------------------------
360 *KARADA
370 IF X+VX>607 OR X+VX<0 THEN VX=0
380 IF GM=1 THEN SW=SW-1:GOTO 400
390 X=X+VX:Y=Y+VY:IF Y<335 THEN VY=VY+.1 ELSE Y=335
400 IF GM=1 THEN PUT(X,Y),G%(0,5),PSET ELSE PUT(X,Y),G%(0,AN+1),PSET
410 IF SW<0 THEN GF=0
420 RETURN
430 '----------------------頭-------------------------------------
440 *ATAMA
450 IF AT<>0 THEN GOTO 480
460 AX=X:AY=Y-30:RS=0
470 GOTO 560
480 IF AX<1 OR AX>605 THEN AVX=AVX*-1
490 IF AY<>335 THEN AX=AX+AVX:AY=AY+AVY
500 IF AX+AVX>1 AND AX+AVX<605 AND AY<335 THEN AX=AX+VX/5
510 IF AY<335 THEN AVY=AVY+.02 ELSE AY=335:AVX=0:AVY=0:RS=0
520 IF AY+AVY<16 THEN AVY=AVY*-1
530 IF AX+AVX<1 OR AX+AVX>605 THEN AVX=AVX*-1
540 IF AX+AVX+VX<2 OR AX+AVX+VX>605 THEN AVX=AVX*-1
550 IF ABS(AX-X)<30 AND ABS(AY-Y)<30 THEN AT=0:RS=0
560 PUT(AX,AY),G%(0,0),PSET
570 RETURN
580 '----------------------敵-------------------------------------
590 *TEKI
600 FOR I=0 TO 4
610 ON E(I) GOSUB *APPEAR,*MOVE,*DEAD
620 NEXT
630 RETURN
640 '----------------------敵出現---------------------------------
650 *APPEAR
660 IF EN>=20 THEN EVX(I)=0:EVY(I)=0:E(I)=2:DF(I)=1:GOTO 690
670 EX(I)=RND(1)*604+3:EY(I)=16:EVX(I)=RND(1)*6-3:EVY(I)=RND(1)*3
680 E(I)=2:EN=EN+1
690 RETURN
700 '----------------------敵移動---------------------------------
710 *MOVE
720 EX(I)=EX(I)+EVX(I):EY(I)=EY(I)+EVY(I)
730 IF EX(I)+EVX(I)<0 OR EX(I)+EVX(I)>607 THEN EVX(I)=-EVX(I)
740 IF EY(I)+EVY(I)>335 OR EY(I)+EVY(I)<16 THEN EVY(I)=-EVY(I)
750 IF ABS(EX(I)-AX)<30 AND ABS(EY(I)-AY)<30 AND DF(I)=0 THEN E(I)=3:GOSUB *HATAMA
760 IF GM=0 AND ABS(EX(I)-X)<30 AND ABS(EY(I)-Y)<30 AND DF(I)=0 THEN SW=10:GM=1
770 IF DF(I)=0 THEN PUT(EX(I),EY(I)),G%(0,3+EP(I)),PSET
780 EA(I)=EA(I)+1:IF EA(I)>10 THEN EA(I)=0:EP(I)=EP(I) XOR 1
790 RETURN
800 '----------------------敵頭にぶつかる------------------------
810 *HATAMA
820 AVX=(AX-EX(I))/15:AVY=(AY-EY(I))/15:RS=RS+1:SCORE=SCORE+RS*500
830 IF AX+AVX<1 OR AX+AVX>606 OR AX>606 THEN AVX=0
840 IF AY+AVY<16 OR AY+AVY>367 THEN AVY=0
850 FOR J=0 TO 255
860 OUT &H3FDB,255-J:OUT &H3FDB,21-RS: BEEP 1
870 NEXT:BEEP 0
880 EW(I)=10
890 RETURN
900 '----------------------敵　死ぬ-----------------------------
910 *DEAD
920 EW(I)=EW(I)-1
930 IF EW(I)>0 THEN PUT(EX(I),EY(I)),G%(0,5),PSET ELSE E(I)=1:EG=EG+1
940 IF EG>=20 THEN GF=0
950 RETURN
960 '--------------------GAME　OVER--------------------
970 *GAMEOVER
980 LOCATE 32,8:PRINT"Game Over"
990 LOCATE 30,14:PRINT"PUSH SPACE KEY"
1000 IF HISC<SCORE THEN LOCATE 31,11:PRINT"New Record!!":HISC=SCORE
1010 WHILE INP(&HE9)<>191:WEND:RETURN
1020 '--------------------タイトル------------------------------
1030 *TITLE
1040 LOCATE 24,10:PRINT"～　頭　を　つ　か　え　！　～"
1050 LOCATE 32,15:PRINT"push space key"
1060 WHILE INP(&HE9)<>191:WEND:RETURN
1070 '--------------キャラクター作成-------------------------
1080 *MAKECHR
1090 CLS 3
1100 FOR I=0 TO 15:READ A$:COLOR=(I,VAL("&H"+A$)):NEXT
1110 DATA 7,0,F0,FF,F00,F0F,DF5,FFF,777,A,A0,AA,A00,A0A,AA0,AAA
1120 FOR P=0 TO 6:READ C
1130 READ A$,B$:C$=A$+B$:FOR K=0 TO 15:FOR J=0 TO 3
1140 A=VAL("&H"+MID$(C$,K*4+J+1,1))
1150 FOR I=0 TO 3
1160 LINE(J*8+(3-I)*2+1,K*2+1)-STEP(1,1),-(((2^I) AND A)<>0)*C,BF
1170 NEXT:NEXT:NEXT
1180 LINE(0,0)-(33,33),C,B
1190 READ A$:FOR I=0 TO VAL("&H"+MID$(A$,1,1))
1200 X=VAL("&H"+MID$(A$,I*4+2,1)):Y=VAL("&H"+MID$(A$,I*4+3,1))
1210 C=VAL("&H"+MID$(A$,I*4+4,1)):D=VAL("&H"+MID$(A$,I*4+5,1))
1220 PAINT(X*2+1,Y*2+1),C,D:NEXT
1230 GET(1,1)-(32,32),G%(0,P)
1240  NEXT:RETURN
1250 DATA 1,000003C00FF01FF8399C6006C0038001
1260 DATA   8421800180019819FF947E220041FF8
1270 DATA   08861                                  '頭
1280 DATA 1,3FFC2424342C2424481248124BD2342C
1290 DATA   1428142813C8300C6006C00387E1FC3F
1300 DATA   43661D56185518C518741                  '体1
1310 DATA 1,FFFF9009C8134812481248122BD4342C
1320 DATA   1428142823C44002418242422 1841FF8
1330 DATA   43661D56185519C518741                  '体2
1340 DATA 1,C003A005900997E998195E7A20042004
1350 DATA   23C42424242423C410080FF000000000
1360 DATA   336311171E1718A21                      '敵1
1370 DATA 1,0000000000007E008101E7820042004
1380 DATA   23C4242424246 3C690098FF1B00DE007
1390 DATA   354311E71EE718A21                      '敵2
1400 DATA 7,0100410421081010000000000000E00E
1410 DATA   00000000000010102108410401000000
1420 DATA   10007                                  '消える
1430 DATA 1,FFFF000000000000E007181807E00000
1440 DATA   00000000000000000000000000000000
1450 DATA   111411E61                              '地面
```

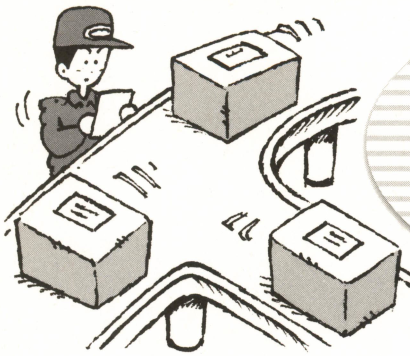
★いくら懐古趣味だからといっても，ディスクシステムのマリオ2を30分でクリアした俺。（東京都足立区・オカッチ19歳）……**【編：**最近の影さんときたら，15年前の投稿プログラムを引っ張り出して遊んでるんですが，懐古趣味？　**影：**ぐはははは，うっしっし。**】**

FX-890P用

リズミカルに荷物を仕分けしよう

いそげ宅急便2001

(たっきゅうびん)



八木　克彦(33歳)

内容

ここはとても忙しいIT宅急便の仕分けセンターです。このセンターでAさんたちは，毎日数え切れない量の荷物をさばき，１個ずつ正確にトラックへ積まなければなりません。ところが，こんなに忙しいのに同僚の２人が，かぜをひいて休んでしまいました。そのため，Aさんはたった１人で荷物を仕分けするはめになりました。とてもとても忙しくなってしまいました。しかし，Aさんは荷物をさばいているうちに大発見しました。よく見るとランダムに流れてきている荷物にはピコピコパターンがあることを発見したのです。ピコピコパターンで流れてくる荷物を，ピコピコリズムに乗って仕分けすれば，つらい仕事も楽しくピコピコ。さあ，あなたもピコピコリズムに乗って，レッツダンス。

遊びかた

荷物(数字 1,2,3)が画面左からベルトコンベアーに乗って流れてきます。荷物と同じ番号のトラックのところへきたら，その番号の数字キーを押すと10点を得点できます。荷物の番号とトラックの番号とを間違えるとスペアMが一つ減り，スペアMが0になるとゲーム・オーバーです。荷物の取りそこねが，10個をこえてもスペアMがひとつ減り，500点得点するごとに，スペアMがひとつ増えます。

《第１表》変数表&ラベル表

・変数	
P	ポイント　得点
TP	トップポイント　最高得点
M	マン　スペア人数
R	ロード　荷物の取りこぼした数
S	ステージB$(S)で扱われるパターンデータ
W	ウェイト 待ち時間。数値が小さいと速い
L	ループ　キー入力に対する制御
A	ループ
C$	荷物の番号とトラックの番号の比較
D$	スタート時のトラック表示処理
K$	キー入力
T$	荷物のパターンデータの加工と表示
B$()	ブロック　パターンデータの定義
・ラベル	
*START	スタート時の画面処理サブルーチン
*PDATA	荷物のパターンデータ定義サブルーチン
*TRACK	トラック表示とスタートデモサブルーチン
*ST	スタート　開始
*MA	メイン　主処理
*LP	ループ　キー入力ループをW回繰り返す
*NX	NEXT　メインループに戻す
*WA	ウエイト　ゲーム・オーバー後の入力待ち
*GA	ゲームオーバー処理
*INK1	INKEY　1が入力された場合の処理
*INK2	INKEY　2が入力された場合の処理
*INK3	INKEY　3が入力された場合の処理
*MEN	残り人数の処理
*TEN	得点の加算処理
*M	マン　残り人数に関する処理
*R	ロード　荷物の取りこぼし処理

操作方法

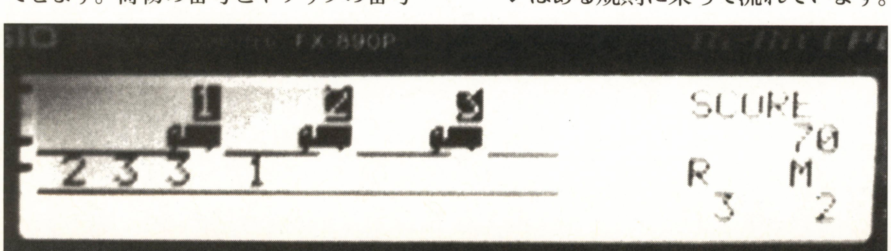
とにかくパターンになれること。パターンはある規則に乗って流れています。パターンを研究して，そしてリズム感をつかんでください。パターンとリズム感を養うと，目をつぶっても荷物をさばけるパターンもあります。

たとえば，画面左から荷物(数字1,2,3)が「1　1　2　2」と流れてきたとき，[1], [2], [1], [2]とテンキーの数字を押せば，うまくクリアできます。パターンは10種類以上あります。

あと，荷物が1のときに[2]や[3]のキーを押すとスペアMがひとつ減ります。あせって数字キーを押さず，落ち着くまで1個や2個はとらず流してしまうことがコツです。荷物を最後まで流しても，10個までならスペアMは減らないので，それまでに崩れかかったパターンとリズム感を取り戻すことです。

プログラムの入力方法

ステージのパターンデータを入力するときに，1020行のREM文を目安にするとデータの入力ミスが減ると思いますので，参考にしてください。あまり役に立たないかも。



《写真１》番号に対応した仕分けを行っていく

CHECKER FLAG

つぐ美：内容は単純なのですが，リズミカルにキーを押していくので，プレイしていて気持ちいいゲームです。

Dr.D：リストはポケコンプログラムの中ではチト長めかな？INK1～3の部分など，共通している動作をうまく一つにまとめていくと，リストが短くなってくるゾ。また，速度向上にもつながってくるしな。

★この季節，レポート課題作成にパソコンが大活躍ですが，気がつくと「ソリティア」なんか，ついついやってしまうところが困りものです。（北海道札幌市・春陽19歳）……【編：息抜きにはいいよね。でも息抜きにならなくなってしまうところがまたミソ。】

改造のアドバイス

パターンデータB$()をいろいろ工夫してください。パターンデータによっては，息もつかせないくらい難しくできると思います。荷物の流れてくるスピードWをあまり速くしすぎると，荷物に追いつけなくなるのでうまく加減してください。

苦労した点

前作の『タイプDEゴーSL』で指摘されたキー入力の反応の悪さは，かなり良くなっていると思います。前作のつぎの作品にFX-890Pで縦スクロール・ゲームを作っていましたが，ゲーム性がもうひとつ面白くなく，いきづまったので保留にしました。そこで，あるとき思いついたこのゲームを作りました。自分でもこれは結構熱くなるゲームと思います。ぞんぶんに遊んでください。

★参考にしたもの

・ゲームウオッチ　マンホール

『いそげ宅急便2001』のプログラム・リスト

```
100 REM -- IT2001 KY ----------
110 CLEAR
120 *ST:CLS:M=3:P=0:R=0:S=1:W=6:DIM B$(10)
130 GOSUB *PDATA
140 GOSUB *TRACK
150 GOSUB *START
160 T$=B$(0)
170 LOCATE 0,2:PRINT T$;
180 BEEP:BEEP:BEEP
190 REM --MAIN----------------
200 *MA:FOR A=0 TO 19
210 T$=LEFT$(MID$(B$(S),20-A,1)+T$,20)
220 IF RIGHT$(T$,1)<>" " THEN R=R+1:GOSUB *R
230 IF R=10 THEN GOSUB *MEN:R=0:GOSUB *R
240 LOCATE 0,2:PRINT T$;
250 L=0
260 *LP:IF L=W GOTO *NX
270 K$=INKEY$
280 IF K$="1" THEN GOSUB *INK1
290 IF K$="2" THEN GOSUB *INK2
300 IF K$="3" THEN GOSUB *INK3
310 WAIT 1:L=L+1:GOTO *LP
320 *NX:NEXT
330 S=S+1:IF S>10 THEN S=1:W=W-1
340 IF W=3 THEN W=5
350 GOTO *MA
360 *GA:WAIT 12
370 LOCATE 30,3:PRINT "0";
380 LOCATE 8,1:PRINT "***GAME OVER***";
390 WAIT 15:CLS
400 BEEP:WAIT 1:BEEP:WAIT 2:BEEP:WAIT 5
410 IF P>TP THEN TP=P
420 LOCATE 10,0:PRINT " TOP SCORE";TP;
430 LOCATE 10,1:PRINT "YOUR SCORE";P;
440 *WA:LOCATE 9,2:PRINT "*PUSH ANY KEY*";:WAIT 7
450 LOCATE 9,2:PRINT "              ";:WAIT 3
460 IF INKEY$="" GOTO *WA
470 GOTO *ST
480 REM --SUB-----------------
490 *INK1
500 C$=MID$(T$,7,1)
510 IF C$="1" THEN GOSUB *TEN ELSE GOSUB *MEN:L=W:RETURN *NX
520 T$=LEFT$(T$,6)+" "+RIGHT$(T$,13)
530 RETURN *NX
540 *INK2
550 C$=MID$(T$,12,1)
560 IF C$="2" THEN GOSUB *TEN ELSE GOSUB *MEN:L=W:RETURN *NX
570 T$=LEFT$(T$,11)+" "+RIGHT$(T$,8)
580 RETURN *NX
590 *INK3
600 C$=MID$(T$,17,1)
610 IF C$="3" THEN GOSUB *TEN ELSE GOSUB *MEN:L=W:RETURN *NX
620 T$=LEFT$(T$,16)+" "+RIGHT$(T$,3)
630 RETURN *NX
640 *TEN
650 P=P+10
660 LOCATE 25,1:PRINT USING "######";P;
670 BEEP:BEEP
680 IF(P MOD 500)=0 THEN BEEP:M=M+1:GOSUB *M
690 RETURN
700 *MEN
710 M=M-1
720 BEEP:WAIT 15
730 *M:IF M=0 THEN BEEP:RETURN *GA
740 LOCATE 29,3:PRINT USING "##";M;
750 *R:LOCATE 25,3:PRINT USING "##";R;
760 RETURN
770 *TRACK
780 DEFCHR$(252)="3C5E7C0C7C7C"
790 DEFCHR$(253)="7C7C7C7E7C7C"
800 D$=CHR$(252)+CHR$(253)
810 D$=" "+D$+"   "+D$+"   "+D$
820 FOR A=0 TO 4
830 LOCATE A,1:PRINT D$;
840 WAIT 3:NEXT
850 RETURN
860 *START
870 FOR A=1 TO 3
880 LOCATE 1+A*5,0:PRINT REV;CHR$(48+A);
890 NEXT
900 LOCATE 25,0:PRINT "SCORE";
910 LOCATE 30,1:PRINT "0";
920 LOCATE 25,2:PRINT "R   M";
930 LOCATE 26,3:PRINT "0   3";
940 DRAW(0,15)-(34,15)
950 DRAW(43,15)-(64,15)
960 DRAW(73,15)-(94,15)
970 DRAW(103,15)-(119,15)
980 DRAW(0,24)-(119,24)
990 RETURN
1000 *PDATA
1010 B$(0)="                    "
1020 REM ***01234567890123456789*
1030 B$(1)="2   1 3  1     2 1  "
1040 B$(2)="  1      2     3  2 "
1050 B$(3)="  1   2   3  2 2 3 3"
1060 B$(4)="  1 1 2 2 1 1   2 2 "
1070 B$(5)="1   2   3 1     2  3"
1080 B$(6)="   2   2 2 2 3 3 3  "
1090 B$(7)="  2  1 1 1    3 3 3 "
1100 B$(8)="1 1 2 2  2 2 2 3 3 3"
1110 B$(9)="2 2 2 1  1  2  2  3 "
1120 B$(10)="    1 1 2 3 2 1 1 1 "
1130 RETURN
1140 REM --END----------------
```


★仲間内で小説を書くのに使っていたペンネーム。大学に合格したら変えるつもりです。（高知県吾川郡・いなかのかちす19歳）……
【つぐ美：で，結果はどうでした？あっ，そうか，つぎの投稿ハガキを見ればいいわけですね。映画の「黄色いハンカチ」みたい。】

CMM試聴室



毎月みなさんの曲を紹介するコーナーです。募集要項をご覧のうえ，どしどしご応募ください。ここで紹介しているデータは，CMMのWebSite（http://www.basicmagazine.net/cmm/）に収録されています。曲への意見，アドバイスなども大歓迎します。

寂寥の街 II

……自作

投稿者：K.USHIDAさん（23歳）

制作音源
SC-88VL

賑やかな中の寂しさを表現してみました

久しぶりに投稿させていただきます。5分半ほどのピアノ曲です。一見「賑やかな」街の「寂しさ」を込めてみました。

データはGM対応です。ベロシティを低く抑えてあるので，音量を大きめにしてお楽しみください。データについて，細かな調整は行っていません。Ballade形式のデータは，ほぼそのまま演奏可能な楽譜になっています。打ち込みのリアルさではなくて，曲そのものの和音の重なりと旋律の流れをお聞きいただければ幸いです。

piano2.mid は1トラックのみ使用，piano2t.midは音程によって定位を変え3トラックになっています。拙作が，多くの方のお耳に入ることを願っております。

編集部から

心が鎮まるようなピアノ曲です。GM準拠の音源で聴くことができます。定位を変えてあるpiano2t.midのほうが，音の広がりは出ていますね。ピアノの打ち込みですが，たしかにピアノ独特の極端なまでのベロシティ（音の強さ）やテンポの変化はなく，粒の揃った音という印象を受けますが，曲の構成がしっかりしているため，完成度は高いと思います。読者のみなさんのお好みで，ピアノの音色やエフェクト付けを変えてみるのもおもしろいでしょう。また，piano2.midを，音域によって定位が自動的に変わるピアノ音色（Piano1wなど，「w」がついているもの）に変えて聴くのもいいと思います。

気になりましたのは，イントロ〜Aメロ部分の既視感（久石譲的？），あと終わりかたです。前者はあまり気にしなくていいと思いますが，後者は，これで曲の後味がだいぶ変わります。不安定な響きで終わらせていますが，その響きのあと，ちょっとだけ安定させた響きで終わらせると，「寂しさ」が「賑やかさ」の裏にあるんですよ，的な，ちょっと後ろに不安定さを隠したような演出ができるかな，と思いました。

BEAT 4 US

……自作

投稿者：松田茂喜さん（48歳）

E-Mail：mazda123@highway.ne.jp

制作音源
SC-8850（SC-88Pro MAP）

1年ぶりで完成させた曲です

正月休みを利用して制作途中になっていた曲を見直し，そのうちの1曲を1年ぶりに打ち込み，完成させました。つまり，貴誌に投稿するのも1年ぶり以上になります。サボっていたせいでショートカットキーは忘れ，ピッチベントの値も電卓を使う始末です。早く勘を取り戻さないと，時間がかかっていけません。

さて曲のほうは，時間を経て聴いてみた自分の作品がなんとも変わっており，また中途半端に終わっていたこともあって，かなり苦労した仕上がりです。なんとなく古めかしさが漂いますね。曲名にある「BEAT」も今の曲からみれば古いBEATかもしれません。

打ち込みで意識したのは，まず「生」。サックスとトランペットはできるだけ人間ぽくなるよう細かくエディットしました。テナーサックスの音色が好きになれないので，アルトをオクターブ下げて使っています。ドラムは曲風にあわせて多少機械的な打ち込みくささを残しております。もともと16分をはねさせており，中盤のアドリブでは結構グルーブ感を出したため，ジャンルが定かでない雰囲気の曲になりましたが，若い人の投稿が多いなか，中年代表として掲載いただければ幸いです。

投稿ありがとうございました

2月1日から28日までに投稿していただいたかたは，以下のみなさんです（順不同）。どうもありがとうございました。データはチェック中ですので，次号以降の試聴室の対象となります。
北海道小樽市・すとらとさんさん，神奈川県藤沢市・楽玄煌さん，宮城県桃生郡・小野寺秀善さん，千葉県柏市・鈴木有人さん，東京都杉並区・satoruさん，山形県米沢市・T.Iwasakiさん，群馬県前橋市・そめっとさん

★《今月のベーマガお楽しみプレゼント》①PS2用ソフト『風のクロノア2〜世界が望んだ忘れもの〜』体験版CD-ROM（製品版はDVD-ROMです）／10名様（提供：ナムコ）　②PS2用ソフト『首都高バトル0』の特製QUOカード／5名様（提供：元気）　③Windows用3DCGソフト

編集部から

久しぶりの投稿ありがとうございます。軽快なビートの上に腰のしっかり座った管楽器のメロディが乗った，明るさと落ち着きが同居している曲です。バッキングのギターの刻みが心地良いですね。中盤のエレピ→ベースのアドリブ部分もなかなかかっこよく決まっています。各音の抜けもよく，聴きやすい曲に仕上がっていますね。

管楽器の打ち込みですが，アタックの雰囲気は出ていると思うので，あとは伸ばす部分を揺らしてあげるとまた雰囲気が変わると思います。エレピも若干揺れがあるといいですね。スネアとタムが左側に寄っていて，これがスタイルなのだと思うのですが，それに対する右側のパートが主にギターの刻みだけという状態で，曲全体が左にかしいで聴こえてしまうのが残念でした。曲の構成はいいと思いますが，管楽器のフレーズ自体に，もうちょっと上下するような遊び心があるといいかもしれませんね。全体的にノリはいいのですが，ノッてるけどおとなしいというイメージもあるので，エレピのあたりでおしゃれさを出すために遊び心を加えてみてください。

old-fashion

………自作

投稿者：新井真樹さん（26歳）

E-Mail：m.arai@aurora.ocn.ne.jp

制作音源
SC-88

ピアノトリオ風に仕上げた曲です

今回はピアノトリオ風に仕上げた曲を投稿します。この曲は4年くらい前にNHKの“趣味悠々”というシリーズで“山下洋輔のジャズの掟”という番組を見て，勉強（？）して作った曲です。この番組は自分にとってジャズの和音やアドリブの作り方など大変参考になりました。一応ビデオも撮ったのですが，とり忘れた回もあるので誰か全部もっている人がいたら連絡下さい（だいぶ前の番組だから無理かもしれないけど……）。

曲名は考えてつけたものではなく，感覚的につけました。自分ではモード的に仕上げたつもりですけど，はたしてそう聴こえるでしょうか？　自信ありません。

曲の構成は2小節フレーズのベースラインの上に，まずメロディを提示してあとは各楽器ごとアドリブでまわして作ってあります。ジャズっぽい勢いがあるノリが伝われば幸いです。ジャズってMIDIで表現するのはやっぱり難しいですね。

編集部から

ピアノの響きとコード進行がとてもおしゃれな曲です。ドラミングも軽快で，ジャズ的雰囲気はとてもよく出ていると思います。ベースとドラムが特にいい感じですね。打ち込みの点では，ピアノはもっとホンキートンク的な音色でもいいかもしれませんね。また，場面はかなり細かくうつり変わっているのですが，変わり目がわかりにくく，ピアノが漫然と鳴っているようにも聴こえてしまうのは残念です。ピアノの音域全体を使うような派手なフレージングも研究してみてください。また，スネアの音量を少し下げて，もう少し中央に寄せたほうが聴きやすくなると思います。逆にいえば，ここまでスタイルづくりができているので，あとはフレージングのがんばり次第ということにもなりますね。

みなさんのオリジナル曲データをお待ちしています！

投稿の注意

投稿は，自分で入力したオリジナル曲で，未発表のものに限らせていただきます。市販のMIDIデータ曲集や，パソコン通信ネット，インターネットのWebSiteにアップされているものを改造しただけのデータはお断りいたします。ただし，パソコン通信，インターネットのWebSiteについては，投稿者本人が公開しているものはOKです。

掲載された曲に関しては，規定の原稿料をお支払いいたします（掲載誌発売後1か月程度かかります）。曲データは，CMMのWebSite（http://www.basicmagazine.net/cmm/）にも，コメントとともにアップロードされます。

CMMのWebSite上，もしくはE-Mailによる投稿も受け付けています。

●あて先・問い合わせ先
〠141-8715　東京都品川区東五反田1-11-15
電波新聞社　マイコンBASICマガジン編集部
「CMM試聴室」係
☎ 03-3445-6111
E-Mail cmm@basicmagazine.net

投稿の内容

下記の3点を封筒に入れ，編集部まで送ってください。

1.曲データの入ったメディア

MIDIデータはフロッピー・ディスク，オーディオ・データはCD-R，MD，DAT，カセットテープ，MOなどでお送りください。曲名と氏名を書いたラベルを必ず貼り付けてください（CD-Rはケースの紙に記載してください）。また，Macintoshユーザーの方は，極力MS-DOSフォーマットのフロッピーで投稿してください。

2.原稿

以下の項目をお書きください。
①住所（郵便番号も忘れずに！）
②氏名（必ずふりがなをつけてください。ペンネームの方は本名も忘れずに）
③年齢
④職業（または学校名・学科）
⑤電話番号
⑥E-Mailアドレス（お持ちの方で，公開してもいいという方はその旨をお書きください）
⑦制作音源，使用ソフト，パソコンの機種名
⑧再生時の注意
⑨データ作成時に工夫したことなど（200字～400字程度）

3.応募カード

本誌綴じ込みのものをご使用ください。コピーしたもの，同一フォーマットで書かれたものでもかまいませんので，必ず同封してください。

E-Mailで投稿する場合

E-Mailで投稿する場合，①MIDIデータ，②原稿のテキスト・ファイルの2種類を，なるべく圧縮ツールで圧縮して一つのファイルにまとめ，メールの添付ファイルとしてお送りください。ファイルが多く圧縮していない場合，取りこぼしがある可能性があります。また，メールの本文に原稿を書くのも取りこぼしの原因となります。

LHAで圧縮する場合，lh6，lh7形式は使わず，lh5形式で圧縮してください。

Outlook Expressをお使いの方は，htmlメールではなく，普通のメールで投稿してください。

その他の注意とお願い

採用，不採用にかかわらず，お送りいただいたフロッピーや原稿はお返しできませんので，必ずコピーを取っておいてください。

MIDIデータは，できるだけ，スタンダードMIDIファイルに変換したものもお付けください。スタンダードMIDIファイルでないものは，WebSiteにアップロードの際，編集部にて変換させていただきますので，ご了承ください。
※本誌もしくはWebSite掲載のデータ，プログラムを使用したこと，または使用できなかったことによるいかなる損失についても免責とさせていただきます。あらかじめご了承のうえ，ご本人の判断でご使用ください。

マイコンBASICマガジン
明日のスター☆プログラマー

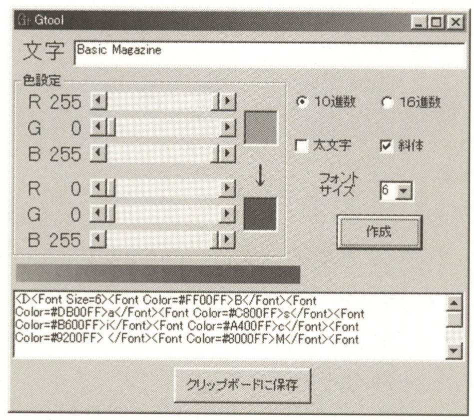
今月のベスト・プログラマー

編：今月から，このページとは別に「Programmer's Room」というページができました。新たに投稿プログラムに対するコーナーわくを広げることで，掲載された作品に対するコメントをより多く掲載できるようになりました。

影：いわば，このコーナー本来の意味合いをより強く打ち出せるようになったわけ。これからは，なるべく多くの意見を載せていくようにするゾ。

つぐ美：それでは，さっそくベスト・プログラマーの選考に移りましょう。わたしは，今月の作品では『BASIC Magazine投稿しようよ。』がお気に入り。このツールによってプログラムの投稿が増えたらウレシイですしね。

編：実用プログラムでは『GTool』も実用性が高そうでオススメですね。HTMLコードでグラデーション文字を作るというツールは，あまり見かけないので着眼点という面でも評価できる作品です。



▲VB用『GTool』

影：変わったところでは，『VBもぐら』はリストも短く，ベーマガの基本コンセプトに沿った作品じゃないかな？

Dr.D：うむ，これらはいずれもVBの作品じゃな。やはりVBプログラムが，今のベーマガでは主流といった感じじゃな。ワシとしても『BASIC Magazine投稿しようよ。』が気になったが，この作品のCHECKER FLAGで述べているようにチト不具合があるのが難点じゃな。

影：では，Dr.今月のベストはどの作品ですか？

Dr.D：編クンがオススメした『GTool』が良いのではないかな？フォーム上のコントロールが多少多いが，リストは比較的コンパクトにまとまっておるし，編クンの言うとおり実用性もあるツールに仕上がっておる。

つぐ美：というわけで，『GTool』の作者，中尾孔一さんが今月のベスト・プログラマーで～す。受賞おめでとうございます。

影さんのマジック・ボックス

ワンテク&アイデア募集コーナー

このコーナーは，みなさんのプログラミングに関するワンポイント・テクニックや，アイデアなどを紹介していこうというもの。投稿する際は，機種や動作環境などもしっかり明記してください。

あて先：〒141-8715　東京都品川区東五反田1-11-15 電波新聞社 マイコンBASICマガジン編集部「影さんのマジック・ボックス」係
e-mail
Webmas@basicmagazine.net
件名に，影さんのマジック・ボックス係と記入してください。

*KAGE*さんのマジックボックス
《Windowsカスタマイズ法-その1》

データファイルを開きたいとき、いちいちメモ帳を起動してそこにドラッグ&ドロップしてファイルの中身を確認する という風にやっている人はいませんか？実は「送る」を利用するともっと簡単にできるのです。

Windowsのシステムがインストールされているフォルダにある Send To フォルダ（普通は、C:¥Windows¥Send To）にメモ帳へのショートカットを作成し名前を「メモ帳」とします。（適当でもいいけど）

開きたいファイルを「右クリック」→「送る」→「メモ帳」（さっきつけた名前）とすると何と選択したファイルが開かれるではありませんか！

これは他のアプリケーションでも応用がきくので、やってみたい人は同じようにしてやってみてください。

兵庫県宝塚市P.N.「KHA」

▲KHA

投稿ありがとう

黒澤信次／T.Y／@SYD／石山明弘／木村雄一／Tomrizer／中尾孔一／吉岡敬浩／八木克彦／チュにひろ／ito-yu／石山明弘／tayosan／T-Soft／富野裕太／谷澤友哉／伊藤隆朗／金井義男／バビブベボーちゃん／西村隆司／幸田夢彦／Natuno／河本幸広／ビッグ・サイトウ／山内卓也／滝沢正臣／池田　暁／辻善喜／GOTO／鮫島みちお／春日　格／亀島諸雄／ション／鈴元也寸志／aifl／wama／大崎敬雄／三上　賢／黒鵜嘉久
※全員紹介できなくてごめんなさい。

投稿プログラム・コーナー

Programmer's Room

編：今回から始まった新しい企画のページ「Programmer's Room（プログラマーズ・ルーム）」。ここは，今まで「スター・プログラマー」のコーナーなどでしか触れられなかった投稿プログラムの状況などをよりくわしくコメントしていくことで，プログラム・コーナーに，もっと身近に参加してもらおうという趣旨のもとで設けられたコーナーです。ページ内容としては，残念ながら採用に至らなかった作品に対して，一言程度アドバイスをしていく「一口CHECKER FLAG」や，今月号の投稿状況を総評という形で報告していく「今月の総評」などのコーナーがあります。これらのコーナーをぜひ，これからのプログラム投稿に役立ててください。

今月の総評

それでは，5月号全体の投稿について。今月は全体的に投稿が少な目。超大作では良い作品が目立ちましたが，やはり基本的にはアイデア重視のベーマガ。シンプルだけどしっかり作られた投稿作品がほしいところですね。また，影さんが期待しているWonderWitchの投稿が少なかったのも残念。しかし，逆に言えば掲載のチャンス。Witchユーザーはどんどん投稿してきてください。

一口CHECKER FLAG

●Visual Basic用『万年カレンダー』／石山明弘（32歳）
つぐ美：ただ，カレンダーを表示させる機能だけじゃ物足りないわね。また，リストファイル（VBだとfrmファイルなど）も一緒に投稿してきてください。

●Visual Basic用『拡張子変換機』／tayosan（17歳）
影：拡張子を変更するだけじゃ，実用性は高いとは言えない。最低でも複数のファイルの拡張子を一度に変更できるなどの機能がほしいところ。

●Visual Basic用『Infinite Shooting』／T-Soft
つぐ美：完成度は高いシューティングだけど，やはり内容が単調。今の状態からゲームとしてどう味付けしていくかがポイントね。

●Visual Basic用『競丘蒸気占い version.1.0』／石山明弘（32歳）
Dr.D：レース結果で占いというアイデアは面白い。じゃが，レーススタートまでに入力する情報がちと多すぎる気がするのぉ。また，画面が800×600でも収まらないのはチョット不親切じゃな。

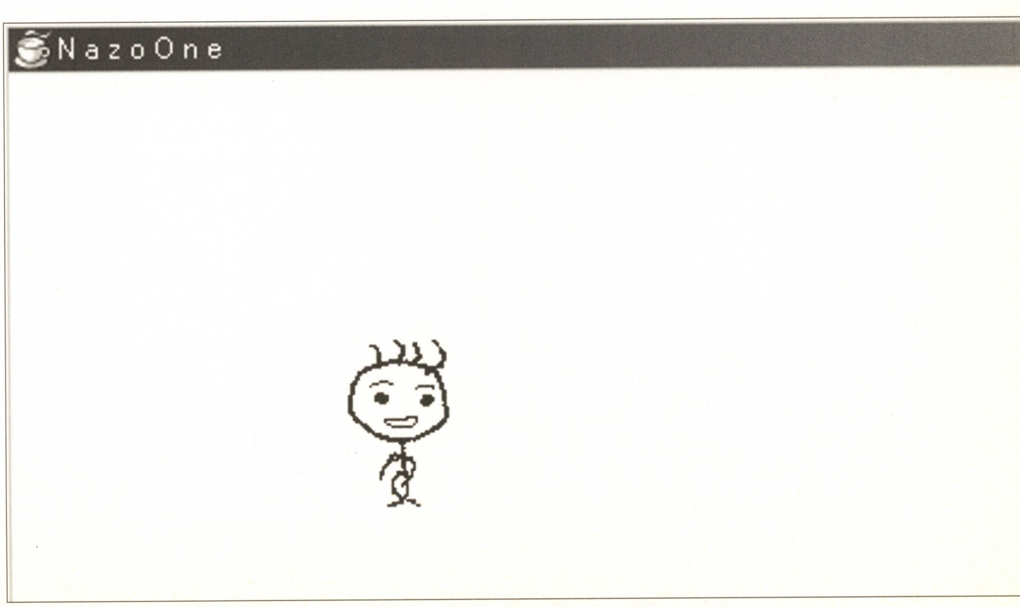
●セガサターン用『おもちゃのチャチャチャ』／KO（40歳）
影：タイトルはおもしろそうなんだけど，ゲームはもぐらたたき。動きも遅くゲームになってない。サターンBASICユーザーがんばれ！

●Delphi5用（超大作）『TS4-Era』／富野裕太（20歳）
つぐ美：テキストのみで書かれたキャラクターはかなりの出来。だけど，テキストシューティングの割には動作が重いのと，動作が不安定なのが難点。実際編集部でチェック中にもフリーズしてしまいました。

●MSX用『海底大戦』／黒澤信次（32歳）
つぐ美：久しぶりのMSXプログラムでしたが，よっぽどおもしろいか，アルゴリズム的によくできていないと掲載はきびしいかも。

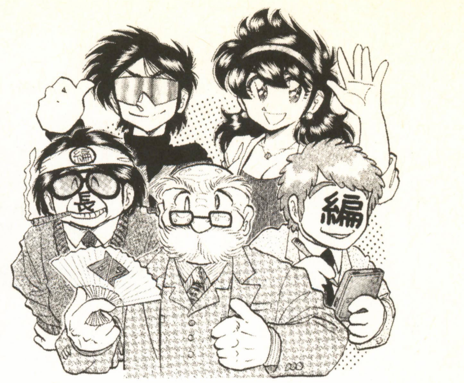
今月はもう一歩でした

●HSP用『ナゾワン』／ito-yu（17歳）
編：一つのキーで，キャラクターを画面外に出さないよう操作していくゲーム。キャラクターの動きにしっかりと加速度がついている点はなかなか。だけど，ゲームとしては，キャラクターが移動する向きを表示したほうが親切ですね。簡単に修正できる部分ですが，この機能があるとプレイが長続きして面白くなってきます。



▶今月はもう一歩でした
HSP用『NazoOne』

あくせすROOM



春の名物と言えば！

脱皮する際に，脂肪のコートまでごっそり棄てちゃった気もしないではないベーマガですが，みなさんから励ましや，お叱りなど，いろんな声をいただきました。何の前触れもなくスリムになった点，お詫びいたします。これからのベーマガはプログラミングを中心に，パソコンで創るクリエイティブ性を前面に打ち出し，その楽しさを多くの人たちに伝えていこうという方針で，がんばっていきますので，応援よろしくお願いいたします。

さてさて，先月は事前に注意し忘れたけど，インターネット世界では，4月1日はその日限定のイベントが多いのです。そう，エイプリール・フールです。

去年は元気が『首都高ラブ』というニセ恋愛SLGの体験版を公開したり（ちゃんと遊べたよ！），キャメロットが透明色のゲームボーイ（完全に透明なので写真にうつっていない！）を発表したり，もうツッコミどころ満載だったのです。

というわけで，読者のみなさんが見つけたウソ情報を募集しま～す。締め切りは4月15日必着の6月号掲載。画面写真もあると，ベターなのであります。



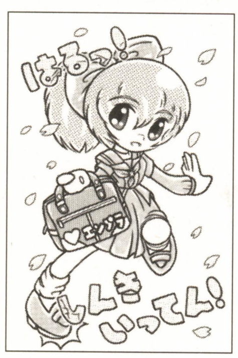
◀千葉県柏市・鈴木有人
つぐ美：青年と少年のはざまを旅する勇者リンク。副作用で幼児化しちゃった？



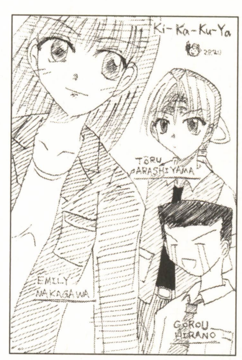
◀大阪府吹田市・ARAI
編：2000は一般ユーザーにはちょっと使いこなせないOSですよね。



◀北海道札幌市・春陽
つぐ美：女の子が見守る前で，ちゃんと飛べるかな？　ドキドキ。



◀愛知県刈谷市・りんす
影：気持ちの切替えが大切だね～。
編集長：早く仕事に取りかからんか！



◀和歌山県橋本市・ヅラChu
編：「読者の闘技場」でさんざボケてくれた登場人物の名前が明らかに。



◀東京都江東区・都田祥
影：技術に自信が足りなければ，アイデアやシチュエーションで勝負！

さようなら闘技場～



◀愛知県豊田市・邪道坊や
影：『タイピング・ワープ』発売日には，数十人の森雪が秋葉原を練り歩いてたぞ。

MebiusLinkに捧げる



◀埼玉県本庄市・MINO
編：さようならの別れがあるからこそ，新しい出会いもあるのですよ。

スキマ企画
萌えっ子プロダクション
（愛称：萌えプロ）

影：なにを描いたらいいのか思いつかない悩めるイラスト投稿者のために，テーマを提供することにした。毎月私がちょっとしたキーワードを三つ並べるので，それらを満たした魅力的なキャラクター（性別は問わない）を表現してほしい。

このキーワードというのは，使いかたしだいでは，単体でも圧倒的な破壊力を持つ危険な必殺兵器ばかり。その扱いにはくれぐれも注意するように。

記念すべき初回のテーマは「しっぽリボン」「オーバーニーソックス」「帽子」だ。まあ，解釈は各自にまかせるが，主旨から外れたダジャレ的なイラストは没になりやすいので注意。諸君の萌え属性に期待する。

7月号のお題キーワード
- ●しっぽリボン
- ●オーバーニーソックス
- ●とりあえず帽子

締切は4月30日必着

今月の大ボケ



◀大阪府吹田市・ARAI
影：機械仕掛けの帽子屋が怖いよ～。
編：マシンパワーがかなり必要です。

学生さんいらっしゃい

♥今度，卒業旅行で電波ビルに行こうと思いますが，社内見学などはやっているのでしょうか？
（京都市・**リョウ　15歳**）
つぐ美：春から嬉しい高校生。でも残念。いまは警備が厳しくて，なかなか編集部の見学はできないの。
影：ライターになれば話は別だよ。

♥私の借金は日経育英奨学会から借りてる専門学校の学費だけですよ。卒業すれば返さなくていいらしいけど，ちょっと不安です。朝早いです。
（福島県いわき市・**さーがろき**）
編：新聞配達ですね。
影：早起きのコツは，早く寝ることだ。
編：そうでもない人も多いですよ。

♥小学校のころは残ったデザートはじゃんけんでした。
（秋田市・**福田　拓　22歳**）
つぐ美：給食のあまりもの争奪戦。勝って喜んでいたら，隣の教室から「こっち足りないけど余ってる？」と悲しい問い合わせがあったりして……。
影：ゆで卵に上手に穴を開けて，ハシでほじくり出して，すっかり食べ終わったやつを配膳台にもどしたもんだよ。
編：お，あまってるじゃーん（グシャ）っていうイタズラですね。
つぐ美：ミカンでやった記憶が……。

♥密かに「高専生」の欄が追加されてますね。多いんですか？
（熊本県菊池郡・**挙動不審　16歳**）
編：多いです。さすがベーマガ。

OOF ダブルオーエフ

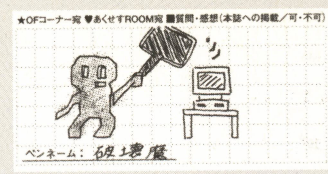
♥ポストカプセルがウチにも届いた。入っていたものは，『エクセリオン』のチラシだった。相当，好きだったみたいです。
（愛知県知多郡・**四谷一獲　25歳**）
影：タイムカプセルと言えば，電波新聞社の倉庫には，昔の投稿プログラムなんかも大量に残っているのだけれど……磁気が劣化して，ほとんど読めなくなっているのだ！
編：5年もたてば，ほとんどの磁気テープは，リードエラー続出ですよね。

♥以前どこかの雑誌で江守徹さんが，パソコンではじめて『ティル・ナ・ノーグ』をやって，その緻密な設定に関心してやみつきになったと話していたことを覚えています。『ティル・ナ・ノーグ』のことは，一度江守さんに取材をしてみてはいかが？
（東京都中野区・**匿名女性　30歳**）
つぐ美：江守さんというと，かなり年輩の俳優さんね。ドラマ「美味しんぼ」の海原雄山役で有名な人かな？

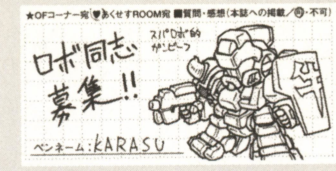
♥アンケートのインターネット歴11年以上ってのがありますが，11年前にインターネットってありましたか？
（埼玉県岩槻市・**大山修史　？歳**）
編：1969年には，原型のARPANET（軍事用）が米国にあったそうです。インターネット接続手順のTCP／IPが無料で公開されたのは1973年ですね。
影：でも，日本で最初のホームページは1992年制作……。

♥3月号のOOFのIWACHAN様お友だちになりましょう。私の闘技場のネタどうですか？　P.N.の由来は，ナムコでGM作ってた小沢純子さんのペンネーム「ZUNKO（ずんこ）」の男の子版です。川本真琴のアルバムとシングル3／3です。ゲットせよ。
（鹿児島県川内市・**ずんた　27歳**）
編：友達募集は，官製ハガキのほうがたくさん書けて便利ですよ。ハガキのすみに電子メールアドレスなぞ書いておくと安全便利です。

♥最近，つけペンとか買って絵の投稿に熱心だったりします。
（鹿児島県加世田市・**IWACHAN　15歳**）
つぐ美：オススメの投稿画材があったら，みんなでレポートしてね。



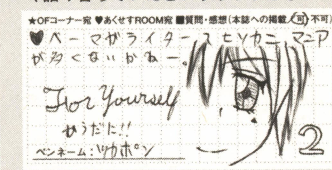
▲京都市・破壊魔　19歳
編：これは，iアプリ『ピコポン』のキャラですね。味があっていいです。



▲東京都新宿区・KARASU　20歳
影：読者集会で，ロボットについてアツく語り合っていたというウワサだぞ？



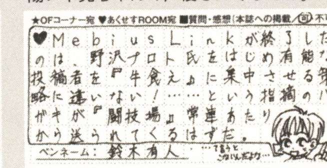
▲北海道江別市・青歌　16歳
Dr.D：それだけの日数があれば，今からはじめても，十分に間に合うぞ！



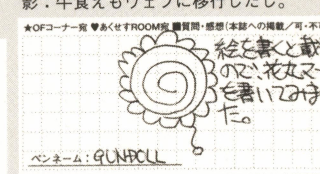
▲福井県武生市・ツカポン　16歳
編：「For Yourself」を歌っているのは，陽ノ下光ちゃんの声優さんでしたね。



▲大阪府吹田市・ARAI　18歳
編：大学の休みが多いのは，一生涯の友人を見つける場所だからなんだよね。



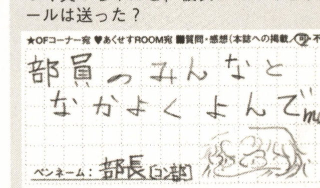
▲千葉県柏市・鈴木有人　28歳
編：案の定，来ませんでした。
影：牛食えもウェブに移行したし。



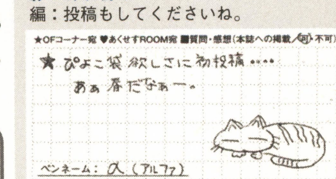
▲滋賀県八日市市・GUNDOLL　32歳
影：載せてしまった……。



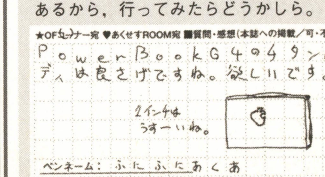
▲兵庫県相生市・ひろくん　32歳
つぐ美：ちゃんと，彼女におめでとうメールは送った？



▲長崎県北松浦郡・部長［コン部］14歳
影：1人1冊！
編：投稿もしてくださいね。



▲京都市・α（アルファ）　17歳
つぐ美：京都市にもゲーマーズのお店があるから，行ってみたらどうかしら。



▲東京都練馬区・ふにふにあくあ　29歳
影：やわらかそうなペンネームだ。
編：チタニウム合金と対称的ですね。

あくせすROOM応募のルール

●愛読者カードや，官製ハガキ，同サイズの丈夫な紙などでお送りください。

●鉛筆や，色鉛筆，涙滴の多いボールペン画は印刷に不向きなので避けましょう。

●ハガキの周囲5mmには文字などが入らないようにしてください。印刷の関係でタチ切られてしまうことがあります。

●投稿内容は，掲載されたプログラムの感想や，学校でのできごと，一発ギャグなど，なんでも結構です。みなさんで楽しいページにしていきましょう。

〒141-8715
東京都品川区東五反田1-11-15
電波新聞社
マイコンBASICマガジン編集部
あくせすROOM　係

webでも受け付けます。Internet読者ハガキでどうぞ。
http://www.basicmagazine.net/

大人気クリエイター集団「ギガ連射」があれこれ語る

GIGA RENSYA つつうらら

第8回

みなさんこんにちは。ギガ連射のふるさとである札幌はこの時期になると，降り積もって道路脇に固められていた雪が溶けだし，歩道も車道も水びたしになる嫌な季節です。でも，ほんのり温かくなってもうすぐさわやかな季節が巡ってきます。春よこーい！そんな季節にふさわしく，今回はさわやかなすがすがしいテーマでいきたいと思います。さあ，若いうちにいってみっかぁ！

『会議』

会議です。さわやか・・・。

細かいことは気にせずにいきましょう。辞書で調べてみると（ありがちな流れ）「関係者でする相談」だそうです。

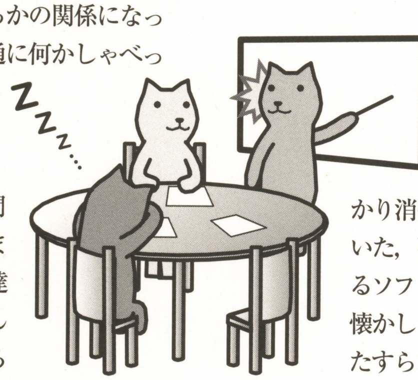
世の中実にさまざまな関係がありますからね。人と人とがお互いを認識した状態ならば，すでになんらかの関係になっています。だから，普通に何かしゃべってても会議？まあ，議題をもって話し合うといったところでしょうか。

ギガ連射もメンバー間で頻繁に会議をしています。ネットワークの発達した現在，チャットなんかで会議する人達もいるようですが，ギガ連射は毎回お互い顔を合わせての，朝まで生会議状態で，終わるころにはヘトヘトになっていることもあります。会議に集まった時点でヘトヘトになっている人もたまにいますが。

どの企業やグループでも当然のように，何かを決めるために会議は重要なものですが，同じようにギガ連射がモノを作り上げていくためには，会議は必要不可欠なものになっています。

では，そこではどのようなことが執り行われているかといいますと，いたって普通ですが皆で話し合い，知恵を出し合い，悩み，苦しみ，たまに憎しみあい，ときには晩御飯のおかずをおすそわけしたりと，内容は非常に濃いです。そんな会議ですが，始まるまでが一苦労。全員集まったからといって，すんなり会議がスタートするはずはありません。ではどうなるかというと，ありがちなパターンとしては，「とりあえず雑談」です。

日ごろのくだらない出来事から，政治情勢に至るまで，内容はさまざま。まあ，雑談ですからね，当然のことなんですが。この雑談のおかげで会議の開始が数時間も遅れることがあります。その他のパターンといいますと「とりあえずゲーム」です。すっかり消費しきって忘れかけていた，引出しの奥に眠っているソフトを引っ張り出しては，懐かしさに心奪われながらひたすらプレイ。このゲームはここが良かった，ここは悪かったなど文句をつけつつ，しまいには「おもしろくねえよ！コレ」と他のソフトに差し替えたりと，せわしな～くプレイ。そして，全ステージクリア。



そんな感じで，一通りすることがなくなると，誰かの「会議でもするか～い？」という力のない声で，会議がスタートするのです。

『弁解？』

このままだとギガ連射はやる気のない，いい加減な団体だと思われてしまいそうなので弁解を。というか弁解ではなく，この一見やる気のないような態度でいることにも理由があるのです。

簡単に言うと，おもしろいモノを作るには自分自信がおもしろがらないとダメ，ということです。

「さあ，おもしろいものつくるぞ～」といって紙に向かったとしても，必ずしもいいアイデアは浮かんでこない。自ら遊んで，楽しんで，文句つけてこそおもしろい発想の肥しとなるのです。

だからギガ連射のメンバーがいつも思っていることは「自分だったらこうする」という発想。既成のモノであっても自分達のアイデアで，より良くできるのではないかと，たえず考えを巡らせているのです。そういう人達が集まる会議の場で雑談したりゲームしたりすることは，共通のものに対してお互いの意見をぶつけ合う絶好の機会なわけです。

で，突飛な意見が飛び交う中，会議でそれらをまとめあげ，ギガ連射のソフトに反映されていくというわけなのです。

ですから，ギガ連射の会議は会議自体がメインなのではなく（会議ももちろん重要ですが）その前の「遊び」の時間が，最も大切な部分なのです。スゴイよ！

はたしてギガ連射の会議のありさまを実際に見て，今言ったようなことを想像できる人がいるのかどうか疑問に残りますが，これは事実です。まぎれもなく事実なのです。

ほんとだってば。

感想・質問・交際の申し込みなど，みなさんからのメールお待ちしています。
otayori@giga-rensya.com

ギガ連射WEBサイト
http://www.giga-rensya.com/

コナミ流。

■(株)コナミミュージックエンタテインメント　〒150-0043　東京都渋谷区道玄坂1-12-1■

うわー，また春が来たよー。社会人になってからというもの，時間が経つのがえらく早くて戸惑います。オイラ，このままで大丈夫か？　なにが？　え？（壊れ気味）

ある意味ヒーリング系

というわけで，「幻想水滸伝Vol.2」のサントラです。幻想の曲を聴く度に，ゲーム音楽も進化したなぁ…と思ってしまいます。今回もそんな1枚。

ゲーム中ではあまりよく聴けなかった音も，CD用に音質調整して，より高い完成度で気持ちよくお聴きいただけるものと思います。ジャケットも,Vol.1の暗めの雰囲気とは打って変わって，お坊ちゃまの優しげな微笑みが嬉しい描き下ろしです。

安らかなひとときを，あなたに…。



幻想水滸伝Vol.2
～クリスタルバレーの決闘～
ORIGINARL SOUNDTRACK
KMCA-97 ¥2,548（税込）
4.4（水）発売中

Z.O.Eだぞい
（おやじギャグ）

きゃー，出ますよ出ますよ。Z.O.Eのサントラが出ますよー。シックでメロウ，かつアグレッシブでイリュージョンでパワフルなサウンドがめいっぱい入ってるサントラですよー。セルヴィスちゃん無事でよかったね，の感動的なエンディングテーマソング2曲も入っていますよー。っていうか，エンディング2曲あるってみんな知ってる？

あとあと，オープニングの歌とかもちゃんと入ってるし，ジャケットなんかはメカもキャラも描き下ろしだし，金色だし，豪華。おじちゃん，商売あがったりだ。



Z.O.E
ORIGINAL SOUNDTRACK
KMCA-98　2,548（税込）
4.25（水）発売

花咲く乙女3人

春全開ってことで，ときめきメモリアル2も花を咲かせちゃいました。そう，あのキャラクター別マキシシングル「Blooming Stories」，こんどは黄色組の元気娘・茜，ほむら，メイが登場です。

「この娘たちのことは，あんまよく知らないんだけど…でも実はずっとお話したいなって思ってたんだ」な～んていう方のために，この娘さんたちを特別にご紹介しちゃいまぁーっすぅ！



ときめきメモリアル2
Blooming Stories4
一文字　茜
KMCM-4　¥1,500（税込）
4.4（水）発売中



ときめきメモリアル2
Blooming Stories5
赤井ほむら
KMCM-5　¥1,500（税込）
4.4（水）発売中



ときめきメモリアル2
Blooming Stories6
伊集院メイ
KMCM-6　¥1,500（税込）
4.4（水）発売中

★一文字　茜（いちもんじ　あかね）さん／家計を助けるためバイトに励むがんばり屋さんです。家事全般が得意なので，将来はきっといいお母さんになるでしょう。

ただし，お兄さんはあの番長さんなので，それなりの覚悟は必要です。でも強く優しくをモットーにいけばきっと大丈夫！　そう，あとは愛があれば!!　詳しくはBlooming Stories4をお聴きください。

★赤井ほむら（あかい　ほむら）さん／我らがひびきの高校の生徒会長さんです。楽しく遊ぶことがと～っても大好きな娘さんなので，体力が続く限り一緒にいろんな所に行ってあげてください。普段はちょっと女っぽさに欠けるところもありますが，仲良しになるとたまに可愛い一面を見せてくれるようになります。詳しくはBlooming Stories5をお聴きください。

★伊集院メイ（いじゅういん　めい）さん／天下の大富豪，あの伊集院家のお嬢様です。ご家庭もそうですが，性格，感覚，すべてが庶民離れした方なので，独学で帝王学を学び，メイ様に庶民感覚を根気よくやさしく教え，ボディガードとしての特訓を積む，など，日々のたゆまぬ努力が大事です。詳しくはBlooming Stories6をお聴きください。

ときめきメモリアル2で遊んだ後は…

そう，「ときめきメモリアル2　Substories～Leaping School Festival～」と「ときめきメモリアル2　対戦ぱずるだま」で遊んだ後は，ゲームサントラも聴いておくれよ！　ジャケットの目印は自転車に乗ったとってもキュートで可愛い光ちゃんだよ。



ときめきメモリアル2
Substories
～Leaping School Festival～
オリジナル・ゲーム・サントラ
KMCA-99　¥2,520（税込）
4.25（水）発売

いわきちですよ21
ミレニアムエディション

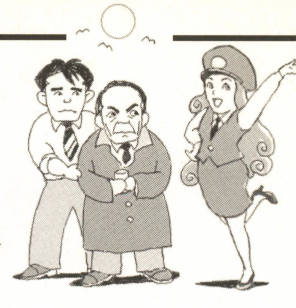
1回だけのつもりがずっと入ったままの「ミレニアムエディション」。さすがにこの時期までくるとカッコ悪っ！　でも面白いからいいや。

でもって，最近のコナミミュージックさんは，いつものことながら大わらわ。9月ぐらい目指して，ドキドキしながら日々を送ってます。ん？　オイラだけか？　マジ？　誰か助けて！　社内，風邪が蔓延してるし。あ～花見行きたい。

http://www.konami.co.jp/kme/

コーエー番デカが贈る

コーエー窓際デカ最前線



横浜の片隅にひっそりと棲息する，窓際デカ長とその部下達。
その任務とは…謎のゲーム会社コーエーの悪事を暴き出し，事件を未然に防ぐことだった！

株式会社コーエー
http://www.koei.co.jp/

「武将を斬ったときの味わい深さを，追求してほしい」（東京都 Fさん）

デカ長：何が楽しくて斬らなきゃいかんのだ。人手不足の世の中だってのに，上からの命令に従って，したくもない英断を………。
刑事3：さりげなく物騒なことを仰有ってるような気がしますが。斬らなきゃならないモノでもあるんですか？
デカ長：やかましい！お前が切れ者とよばれる人間だったらなーんの問題もないんだよっ！

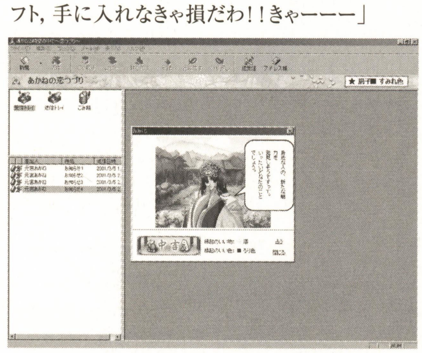
報告書

初めて恋文をしたためたのは，中学生の頃だった。いつも彼女の姿を目で追いつづければ1日1回は目が合うし，登校途中わざわざ彼女の家の近くをとおって物思いにふけっていると，いつも猛スピードで私を追い越しながら顔を見せてくれるし。彼女も絶対自分のコトを好いてくれていると思っていたワケだ。で，「好きです」と想いをしたためたのだが，ま，結果は気にしないで欲しい。ここで言いたかったのは，いわゆるラブレターを出す醍醐味はなんぞや，ということ。それは家族に見られること必須でも郵送するべきか，体育館の裏に呼び出していさぎよく手渡しか，とか，どうやって渡すかを1か月以上は悩み続けることだったりするんだな。だのに，今はメールでお手軽カンタンぴょんで，一発送信とはなんじゃこりゃあ！こんなんでいいのか？！学生諸君!!!

「まあまあ，そんな激昂しないで。おじさんは知らないかもしれないけど，今はメールの世界にもめくるめく浪漫が広がっているのよ。これこれ『遙かなる時空の中で～恋つづり～』はね，プレイステーション版で女の子に大人気の恋愛アドベンチャーゲーム『遙かなる時空の中で』のメールソフトなんだけど。なんと，メールを遙かのキャラとゲーム中の絵をつかって，漫画風につくってしまうことができるのでーーす！

遙かキャラは，あの見目麗しい男性キャラの「八葉」たちや，鬼の一族や，藤姫が使えるんだけど「今宵は楽しいひとときだった…」なんてなんて八葉たちに言わせちゃったりなんかしたら，きゃーーーー♥って感じでしょ？メールのやりとりも華やかになるってもんよね！あと，お気に入りキャラを設定すると，メール着信のお知らせとか，着信内容の報告なんかをしてくれるんだけど，一番嬉しいのは，おみくじをひかせてくれるってこと。その日の「縁起のいい色」や「縁起のいい物」なんかを教えてくれるのよ。

乙女の心を鷲掴み！な上，「GAMECITY」では「恋つづり体験版」を公開中！そして，送る相手がこの恋つづりを持っていなくても，HTMLに対応したメールソフトを持っていれば，ばっちり受け取ることができるから，もう世界中の女の子みんなに送っちゃいたいわね♥絶対このメールソフト，手に入れなきゃ損だわ!!きゃーーー」



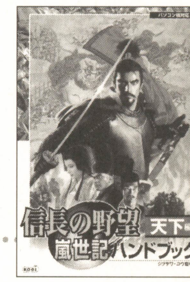


なになに？「縁起のいい物」は酒だって？？も，もしやこのおみくじは未だかつてないほどの信憑性の高い，すぐれものなのか？！そうだ，きっと絶対，そうに違いない。というワケで，これから飲み屋にレッツごー!!

デカ長：顔が赤いぞ，こらぁー！
刑事3：いや～，若き日の恋心を思い出して，赤面してるだけですよ～

今月の押収品 ①

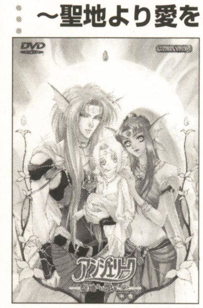
発売中の
『信長の野望 嵐世記 ハンドブック 天下編』



信長の野望ファンのバイブルですか。私にも信じるモノが欲しい今日この頃…。

今月の押収品 ②

発売中のOVA
『アンジェリーク ～聖地より愛をこめて～ 中巻』



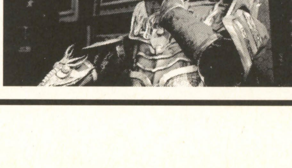
まーかわいらしい子供さん。聖地にはほんっと愛が溢れてるんでしょうね。

第25回 婦警さんの㊙情報部屋

はぁぁぁ～い♥　うぅ～ん，春よネ～♥♥♥　道ばたの桜の匂いにちょっとウキウキ♥　窓辺の日差し思わずウトウト♥な，婦警さんでぇ～す♥　フレッシャーさんたちが歩いてるのを眺めるのもスキなのよネ♥　うふふ♥　みんなガンバって～♥

さてぇ，今回も捜査の中心はPS2の超大作『決戦Ⅱ』!！ついに発売されました～！きゃー♥♥♥　もうクリアした人もきっといるわよネ♥　今回の内容はかなりファンタジック！テーマは「三国志」だけど，あんまりカンケーありませぇ～ん♥　個人的には孔明の奥さん・黄月英(コウゲツエイ)さんが発明するトンデモ兵器が大好き♥♥

火炎放射器やら大魔神やら♥♥　それは実際にプレイしてビックリしてみてくださ～い♥　さらに!！今回は大スクープ！なんと，最初にプレイする「劉備編」の他に，曹操が中心の物語「曹操編」があることが発覚したんでぇ～す♥　婦警さんお手柄～♥　ちょっとクールなオジサマ♥曹操の活躍を楽しみにしててネ～♥　感想もお待ちしてまぁ～す♥　ホントに面白いわヨ♥　婦警さんでした～♥♥



善良なるベーマガ読者の諸君，髭面武将などなんでもいい，コーエー近辺で怪しい人物を見つけたら，その人相書きを送ってくれたまえ。採用者には謝礼として特製テレカを用意している。宛先はこちらだ。

〒223-8503神奈川県横浜市箕輪町1-18-12　（株）コーエー　広報部「ベーマガ目安箱」係

システムソフトの
エンスー学館

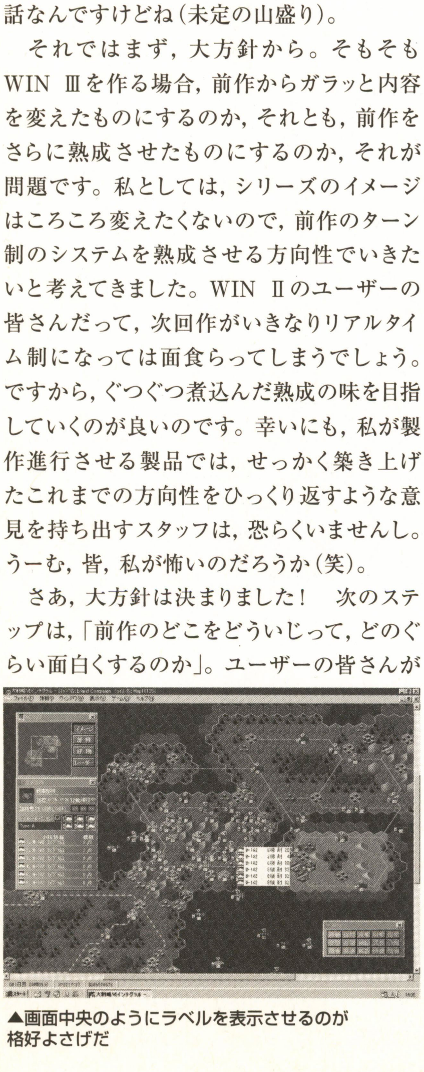
〒810-8665
福岡市中央区天神3-10-30
http://www.systemsoft.co.jp/

「メイキング・オブ・大戦略WIN Ⅱ」第5話
～大戦略WIN Ⅲのあしたはどっちだ!?～

こんにちは，都筑区です。さて，今回は『メイキング・オブ・大戦略WIN Ⅱ』の最終話前編。これまでWIN Ⅱの裏話にお付き合いいただいたからには，きたるべき新作『大戦略WIN Ⅲ』についても聞いていただかねばならないでしょう。とはいっても，WIN Ⅱの制作中に沸いて出たアイデアのお話なんですけどね（未定の山盛り）。

それではまず，大方針から。そもそもWIN Ⅲを作る場合，前作からガラッと内容を変えたものにするのか，それとも，前作をさらに熟成させたものにするのか，それが問題です。私としては，シリーズのイメージはころころ変えたくないので，前作のターン制のシステムを熟成させる方向性でいきたいと考えてきました。WIN Ⅱのユーザーの皆さんだって，次回作がいきなりリアルタイム制になっては面食らってしまうでしょう。ですから，ぐつぐつ煮込んだ熟成の味を目指していくのが良いのです。幸いにも，私が製作進行させる製品では，せっかく築き上げたこれまでの方向性をひっくり返すような意見を持ち出すスタッフは，恐らくいませんし。うーむ，皆，私が怖いのだろうか（笑）。

さあ，大方針は決まりました！　次のステップは，「前作のどこをどういじって，どのぐらい面白くするのか」。ユーザーの皆さんが一番気になるのは，やはりこの点ですよね。でも，焦ってはいけません。ゲームの内容をいくつかの視点に分けてじっくり考えてみましょう。戦場となる「マップ」の構造，兵器ユニット，戦場でのルール，CPUのアルゴリズム，その他エディタやLAN対戦機能などなど。現状（前作）の把握を含め，大きな視点から小さな視点に落とし込んでいくのが肝要であります。


▲画面中央のようにラベルを表示させるのが格好よさげだ

＊　＊　＊

大戦略といえば6角形のマップ，6角形といえば大戦略のマップというほど定着したのが，この戦場マップですね。近年では，6角形のガイドをマップ上に表示したり消したりできるようになりましたが，この先も6角形のガイド表示が仕様から落ちることは決してないでしょう。新作でマップの構造に手を入れるとしたら，「広さ」にこだわってみたいものです。48×48ヘックスの現在のマップに加えて，64×64ヘックスと96×96ヘックスの，2種類の広さのマップが導入できると楽しさ100倍。戦場が広くなればその分，マップ上でのトリックの設定も複雑なことができそうですものね。

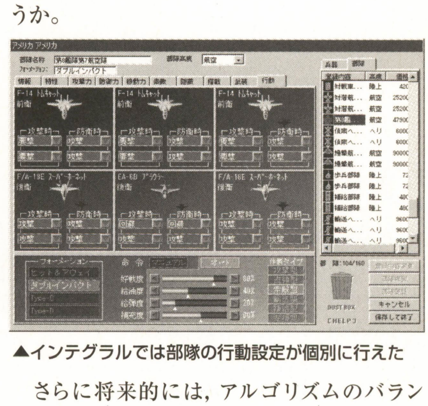
また，建物には耐久度を設定し，占領に少し手間がかかるようにしたり，逆に軍事費や特定の兵器ユニットを使用して建物の耐久度アップさせられるようにするというのも，また一興かと思います。

マップが広くなれば，新たに空中給油機や補給艦などの兵器ユニットを追加しなければなりません。建物に関するルールが変われば工作車や工兵部隊も必要になるでしょう。このように，必然性あっての新兵器導入というのは，なんとも気持ちの良いものですね～。ちょいと余談ですが，工作車や工兵があればトーチカ，レーダーサイトの建設が可能，というのも面白いなぁとも考えてみたりして。

地名や部隊名がマップ上に表示できる「ラベル」機能は，マップ上にベタで文字を表示するのではなく，『大戦略Ⅵインテグラル』のようにヘルプチップ的にスマートに表示させたいものです。

＊　＊　＊

そして，肝腎要のCPUのアルゴリズム……。CPUのアルゴリズムは何種類か用意できると良いですね。たとえば，「いけいけの戦闘型」「占領重視の金儲け型」「地味な動きだが手堅い防御型」「バランス型」など。これらのアルゴリズムを国によって使い分けると，生産タイプの選択ともあいまって，なかなかの個性が発揮できるのではないでしょうか。


▲インテグラルでは部隊の行動設定が個別に行えた

さらに将来的には，アルゴリズムのバランスをエディタで調整できるところまでいってしまうのが良いかと思います。『大戦略Ⅵインテグラル』では，リアルタイム制ということもあって部隊ひとつひとつに同じような概念の設定を行いましたが，ターン制では部隊単位ではなく「国の方針」的な設定のしかたが扱いやすそうです。そんなことも頭の隅に置いておきながら，次回の最終話後編を読んでくださいね。　【次回へ続く】

SEGAのヘンな話題やちょっとお得な情報ならおまかせ!!

ベーセガ

マイコン BASIC
SEGA OFFICIAL
VARIETY SPACE

奈美　(ばたばたばたばたばたー)

竹崎　おっ，奈美さんがなにやら走ってくるぞ…。春はゲームもいっぱい発売される，忙しくも楽しい季節だよねー。

奈美　たけざきさーん！

竹崎　どうしたどうした，またなんか事件かい？ ハイハイ，それはそうと読者の皆さんにちゃんとご挨拶しようねー。

奈美　はぁはぁ，どもも。ゲーマー奈美でぇす。テレホタイムには「ファンタシースターオンライン」をプレイしている，2＊歳のOLでーす！ 以後お見知りおきを～☆よろしくねっ。きゃー，人前に出るなんて，奈美，恥ずかしいぃぃっ。

竹崎　(こ，コイツはなんか拾い食いでもしたのか…!?)。ところで奈美さん，なんか俺に用があったんじゃ。

奈美　あっ，そうそうそうでした。さっき決定したばっかりのニュースを皆さんにお知らせしちゃいますね！ じゃじゃーん，その名も「GameJam」!!

竹崎　あっ，知ってる知ってるー！ 4月12日に発売されるDCのソフトだよねー？ 野球とかサッカーとか人気スポーツのおいしいとこだけを詰め込んだ，簡単操作でわかりやすーい，まるで，「筋○番付」みたいなゲームでしょ？

奈美　そうそうそう！ ルールも簡単だからパーティゲームに，まさにうってつけ……って，ちっがーーーーーーーーーーーーーーう！ それは，「SPORTS JAM」でしょーが!! 私が言ってるのは「GameJam in Zepp Tokyo」ですっっ。

竹崎　…さっきはそんなサブタイトル，ついてなかった気が。あっ，わかった。サターンの時に出たソフトで，ソニックのゲーム4本が一つにまとまったとゆー…。

奈美　そ，それは「SONIC JAM」……。んじゃ，私の方から皆さんに「GameJam in Zepp Tokyo」の実態をご説明しますっ。きたる4月14日の12:00～18:00と4月15日の10:00～16:00に行われるDCのイベントってのが正解です。場所は，お台場パレットタウンのZepp Tokyoですよ。

竹崎　なになに，出展タイトルは「シェンムーII」「サクラ大戦3」「ソニックアドベンチャー2」「クレイジータクシー2」「ファンタシースターオンラインVer.2」「アドバンスド大戦略2001」「アウトトリガー」「SEGA EXTREME SPORTS」「MAGIC：The Gathering」「Dynamic GOLF」「ガンダムバトルオンライン」他多数か，ほう。

奈美　そのほか，ステージイベントも予定されてるんですよ。「シェンムーII」の初公開映像や鈴木裕氏による実機デモンストレーション，ソニックチームの中さんも登場しますし，さらにっ。「サクラ大戦3」の豪華ゲストもお招きしてまーす。

竹崎　これは行かなくてはなるまいっ。

奈美　ハイハイ，竹崎さんはもちろん行くんですよー。仕事でねー。

竹崎　な，なるほど。にしても，やっぱりこのイベント名称は，ちょっとわかりにくくない？ 俺以外にも「SPORTS JAM」や「SONIC JAM」と勘違いした人間が絶対にいるはずだーーー!!

奈美　そんな，夕日に向かって叫ばなくても…。もーしょーがないですねえ，ぷんぷん。仕方がないので，このイベントを企画した部署にちょっと偵察に行ってきましたよ。

竹崎　とは言え，隣の部だから，間にはパーテーションすらないけど…。

奈美　しーーっ。聞き込み調査によりますと，どうやら「GameJam」という名前の名付け親はイベント担当のSさん（仮名）らしいです。

竹崎　そ，それで？

奈美　えと，わたくし，Sさんとコンタクトをとることに成功しましたので，ご報告しますね。以下はそのやりとりです。

奈美　あ，Sさん。突然ですが「GameJam」の名付け親はSさんだとお聞きしましたが？

S　ああ，そうだけど？

奈美　なぜ，この名前にされたのでしょうか？

S　いろんなゲームが集まってる，という意味で。

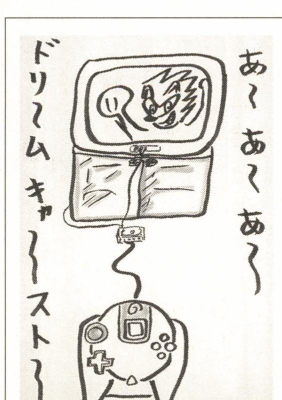
奈美　なるほど。では特に反対意見等もなく満場一致で決定したのですね？

S　うん。

奈美　以上です。このやりとりから察しまするに，イベント名称は，ごくごく自然にすんなりと決定したようです。やはり「SONIC JAM」と間違えるのは竹崎さんくらいのようですねっ。ではでは皆さん，会場でお待ちしてまーす。まったね～！



◀水野夢絵様江 RNA-SIDE



◀レッドドラゴン



◀Aす



◀湯浅裕美

このコーナーを充実させるための，ご意見，お便り，イラストなどなどクビを長くしてお待ちしています。採用の方には特製テレカなど，非売品セガグッズをプレゼント。SEGA竹崎がお便りをすべて読んで，面白いコーナーを作っていきたいと思います。すべてのあてさきは，ここや!!

〒160-0032 東京都港区六本木1-4-30 六本木25森ビル18F (株)セガ カスタマーゲートウェイ部メディアコミュニケーションチーム「竹崎と奈美のベーセガ」係

セガCSパブリシティチームのHPはこちら→http://www.sega.co.jp/dreamcast/cspr/

Falcom® NEWS EXPRESS

日本ファルコム株式会社
〒190-0012 東京都立川市曙町1-14-13
TEL 042-527-6501
FAX 042-528-2714

スタッフ募集中
採用担当BFN係

http://www.falcom.co.jp/





▲新オープニングより。フィーナ＆レアファン必見！

速報1 イースI・II完全版(コンプリート)発売決定！

今月はビッグニュースが二つもあるぞ！その一つがイースエターナルとイースIIエターナルが統合され，ゲーム内容も格段にグレードアップされた「イースI・IIコンプリートバージョン」の発売。新オープニングムービーの追加や新イベント，激ムズ・ナイトメアモードやボス戦タイムアタックモード等々，変更点はこれだけではなく細部も徹底的にこだわってクオリティをアップさせているのだ。さらに，ゲームが本体なのかおまけなのかわからないほどの超豪華な内容なのだ！

■復刻版OVAイース全7巻(DVD-VIDEO)



「I」のストーリーがアニメ化された幻のOVAイース全７巻が待望の完全復刻！

■交響曲イース21stセンチュリー



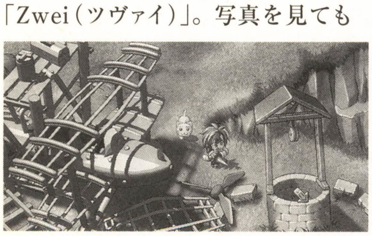
完全新作，新アレンジのイースミュージック史上最高のシンフォニーが完成！

交響曲イース21stセンチュリーのサンプル曲も収録されたイースI・II完全版のデモCDを全国販売店で無料配布中！配布店はファルコムホームページのショップリストに掲載中！
http://www.falcom.co.jp/shoplist/

速報2 完全新作タイトル決定！

XANADU（1985）・・Ys（1987）・・ そして2001年Zweiが登場する。

ビッグニュースの二つ目が，以前から画面だけ公開されていた謎のファルコム完全新作，完全新シリーズのタイトルが遂に決定！タイトルは「Zwei（ツヴァイ）」。写真を見てもわかるように，ほのぼのとしたゲーム画面のアクションRPGだ。これから新情報がじょじょに公開されていくので，ファルコムホームページを見逃すなっ！





FMSが新たに下記販売店で発売開始！

大好評発売中のFMSシリーズの取扱店が増えました。近くの方はぜひご利用ください。

●そうご電器YES札幌
札幌市中央区南２条西３丁目　011-214-2850

●ヨドバシカメラ仙台店
宮城県仙台市宮城野区榴ヶ岡1-3-1　022-295-1010

●ヤマギワソフト館
東京都千代田区外神田3-13-12　03-3253-5111

●ラオックスヒナタコンピュータ館
長野県長野市権堂2210　026-273-2221

●ラオックスヒナタ松本コンピュータ館
長野県松本市双葉19-10　0263-29-1110

Vectorプロレジに「モナモナ」登場！

あの一度ハマったらやめられない「モナークモナーク」が遂にVectorプロレジでも発売開始！リアルタイムで勝手に動いている可愛いキャラに時々命令するだけの，非常にシンプルなルール。立体MAPと時間軸とのタイミングなどパズル要素もたっぷりで，やり込むほどにハマってしまう不思議議なゲームです。プレイしたことがない方はぜひ遊んでみてくださいね。

トクレジ会員になるとさらにお得！

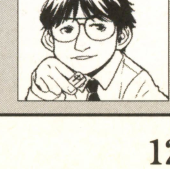
Vectorでは4月から会員制のダウンロード販売開始。会員になると新英雄伝説など，さらに2割引で購入できます。詳しくは，Vectorのホームページをご覧ください。
http://www.vector.co.jp/

プロレジで販売中のファルコム製品を知りたい方は下記までアクセスを！
http://www.falcom.co.jp/vector/

着メロ配信に新曲が追加！

ファミ通ドットコムの着信メロディ配信サービス「Melody Clip」に，新たに新白き魔女の曲が追加されたぞ！曲は「魔獣出現」「小さな英雄-ジュリオとクリスの大冒険」「Win！」「それ見よ我が元気」「ルード城」の5曲。詳しくは下記サイトをチェック！
http://www.famitsu.com/mobile/melomelo/

今度のイースI・IIの特典DVDビデオが全７巻トータル３時間30分！当時の販売価格が１本4,500円だったので，合計31,500円分が無料でもらえることになります。これだけでも買う価値有り！





タイトル冠文：大阪府・木影
タイトル・イラスト：北海道・春陽
本文：暁

入学祝いに
なにわ通信

PRESENTED BY マイコンソフト

http://www.dempa.co.jp/MICOMSOFT/

ADDRESS● 〒530-0005大阪市北区中之島3-2-4
朝日新聞ビル6F 電波新聞社大阪本社内
マイコンソフト株式会社　なにわ通信係

最近，目が疲れ気味です。

先日，目の周りの筋肉が痙攣するので，お医者さんに行きました。

そしたら「眼精疲労です」と言われてしまいました。

左右の視力の違いが原因になっているようですが，マイコンソフトの仕事は，一日中パソコンモニタとテレビを見つめることが多いので職業病かもしれませんね。

きっと，映像機器やビデオゲームなどの開発者たちは，同じような症状で悩んでいるのだろうなぁ，と考えています。

みなさんもテレビの見過ぎには注意しましょうね。

フライヤー募集

前号でもお知らせしましたが，なにわ通信では，フライヤー(ペーパー)を募集しています。募集要項は，以前と同じですが，前回ご参加いただけなかった方のために，もう一度ご説明させていただきます。

1枚の紙に，自分の描きたい文章とイラストを描いてお送りください。

両面印刷したい方は，原稿を2枚お送りください。こちらでの印刷は通常のコピーになりますので，印刷や，紙質，デザインにこだわりたい方は，各自でコピーしてからお送りいただいてもかまいません。

お送りいただいたフライヤーは参加者の方々に配布されます。

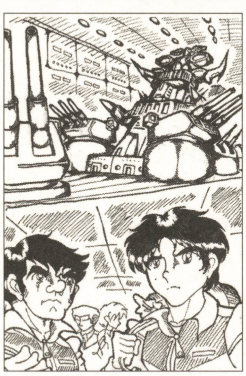
今回は，希望者のために一般配布分も考慮したいと考えておりますので，一般配布可能な方は，そのようにご連絡ください。

一般配布分は，「参加してみたいが，どんな物を描いたらよいか分からないので見てみたい」という方へ配布してみたいと考えております。

また，各自で印刷されてからフライヤーを投稿される際は，参加者分，一般配布合わせて15枚程度お送りいただければと思います。

原稿の締切は，5月末ですが，配布は6月末になってしまうと思います。

ありがたいおハガキのコーナー



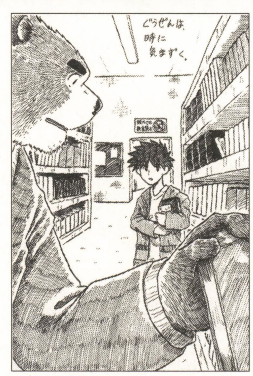
愛知県・邪道坊や
ごぶさたしております。宇宙戦艦，いいですね〜。私の世代は，やっぱり宇宙戦艦ヤマトですね。宇宙戦艦は男のロマンです。



北海道・青歌
10年もピアノを習われていたのですか。
次はどんな楽器にチャレンジされるのでしょうか。



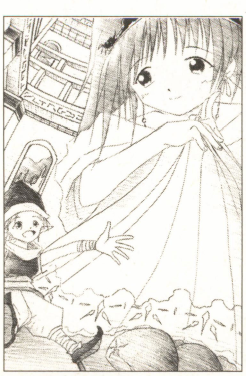
北海道・春陽
旅から帰ってきた少数民族の男性のようです。彼女の愛が伝わってきます。ところで，民族楽器は弾けるようになりましたか？



大阪府・ARAI
熊恋シリーズには，まだ続きが！いつもなら，気軽に挨拶をするはずのシーン。しかし，今は，それさえためらわれる。



大阪府・藤井朱丸
受験勉強お疲れさまでした。今回は，忍者ですね。今度，大昔テレビで放映していた「赤影」が映画化されますね。



大阪府・戦斧王伝説
昔投稿されていたバトルホークさんの大ファンなのだそうです。
あのかっこいい甲冑は印象的でしたね。



宮城県・吏久
銀河鉄道999のメーテルですよね。私，999大好きなのです。SLが宇宙を飛ぶなんて，ロマンがありますよ。



埼玉県・MINO
トーンを使うのが苦手とか。でも，トーンを使わなければ，お金もかかりませんし，ペンの練習にもなります。良いことです。



徳島県・るじパルコ
ぬいぐるみを作る三人姉妹でしょうか。シッポが長いのとか，リボンが長いのとか，いろいろですね。



三重県・中西二朗
このイラストは，中西さんが書かれている。「白い渚で」という小説のキャラたちなのだそうです。どんな小説なのでしょうか。



●復刻版OVAイース全７巻（DVD-VIDEO）
幻のOVA「イース 〜序章・ハダルの章・トバの章・ダビーの章・メサの章・ジェンマの章・ファクトの章〜」が待望の完全復刻！
●交響曲イース21stセンチュリー（完全新作、新アレンジ）
大好評の交響曲シリーズに、イースミュージック史上最高の「イースシンフォニー」ここに堂々完成！
イース
I・II
— 完全版 —
コンプリート
デモCD無料配布中！
オープニングデモムービー＆「交響曲イース21stセンチュリー」サンプル曲を収録！
【特典の値段です！ソフトはただのおまけです！】
【リニューアル版ではありません「Ｉ・II統合のコンプリート版」です！】
【コンプリート版での変更点は…？】
★「Ｉ」の始まりにも、「新オープニングムービー」を挿入！
★上級者用・超激ムズ「ナイトメアモード」を追加！
★物語を楽しみたい方のために「イージーモード」を追加！
★ボス戦だけを楽しむことができる「タイムアタックモード」を追加！
★アドルが喋れるようになった！？
★イベントはもちろん、ゲーム中のグラフィック全体のディテールをさらに強化！
★「Ｉ」のＢＧＭは、原曲の雰囲気を保ちつつ全曲をリアレンジ！
★効果音もよりリアルな音に全て差し替え！
★ＭＩＤＩ再生でフェードアウトや永久ループが可能になった！
…これらの変更はごく一部。細部にも徹底的にこだわりクオリティをＵｐ！
「イースＩ・II完全版」についての情報は、ホームページをご覧下さい。http://www.falcom.com/
■ご購入は便利なオンラインショッピングコーナーをご利用下さい falcom.com（ファルコムドットコム）■社員募集（新卒・既卒・中途）！詳しくはファルコムホームページ「会社案内・社員募集」をご覧ください。
※この広告は新オープニングムービーの画像を使用しています。
Falcom®
NIHON FALCOM CORPORATION
〒190-0012 東京都立川市曙町1-14-13
☎042(527)6501 http://www.falcom.co.jp/

定価550円 本体524円 雑誌18361-05 T1118361050554

第二十巻第五号通巻二二七号 昭和五十七年十二月六日第三種郵便物認可 二〇〇一年四月二十五日印刷 二〇〇一年五月一日発行

電波新聞社 〒141-8715 東京都品川区東五反田一―十一―十五 電話〇三(三四四五)六一一一

2001

わかる！動かせる！
プログラムが組める雑誌

マイコンBASIC Magazine

5

「iアプリ」制作・「隕石避けゲーム」に挑戦
ヒューマノイドロボット「Mk・5」の研究現場を見た！

http://www.basicmagazine.net/

電波新聞社